

802.11 MARKOV CHANNEL MODELING

by

Julio Nicolás Aráuz Salazar

Electronics and Telecommunications Engineering, E.P.N Quito - Ecuador, 1996

MST, University of Pittsburgh, 2000

Submitted to the Graduate Faculty of

School of Information Sciences in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy in Information Sciences with a Telecommunications concentration

University of Pittsburgh

2004

UNIVERSITY OF PITTSBURGH
SCHOOL OF INFORMATION SCIENCES

This dissertation was presented

by

Julio Nicolás Aráuz Salazar

It was defended on

September 21st, 2004

and approved by

Dr. Richard A. Thompson

Dr. Martin Weiss

Dr. Joseph Kabara

Dr. Thomas Savits

Dr. Prashant Krishnamurthy
Dissertation Director

802.11 MARKOV CHANNEL MODELING

Julio Nicolás Aráuz Salazar, PhD

University of Pittsburgh, 2004

In order to understand the behavior of upper layer protocols and to design or fine tune their parameters over wireless networks, it is common to assume that the underlying channel is a flat Rayleigh fading channel. Such channels are commonly modeled as finite state Markov chains. Recently, hidden Markov models have also been employed to characterize these channels. Although Markov models have been widely used to study the performance of communications protocols at the link and transport layers, no validation of their accuracy has been performed against experimental data. These models are not applicable to frequency selective fading channels. Moreover, there are no good models to consider the effects of path loss (average received SNR), the packet size, and transmission rate variations which are significant in IEEE 802.11 wireless local area networks.

This research performs validation of Markov models with experimental data and discusses the limitations of the process. In this dissertation, we present different models that have been proposed along with their validity analysis. We use the experimental data with stochastic modeling approaches to characterize the frame losses in IEEE 802.11 wireless LANs. We also characterize the important factor of current wireless LAN technology, the transmission rate variations. New guidelines for the construction of Markov and hidden Markov models for wireless LAN channels are developed and presented along the necessary data to implement them in performance studies. Furthermore we also evaluate the validity of using Markovian models to understand the effects on upper layer protocols such as TCP.

TABLE OF CONTENTS

1.	INTRODUCTION	1
1.1.	CURRENT STATE OF CHANNEL MODELING OF WIRELESS CHANNELS	1
1.2.	ISSUES RELATED TO CHANNEL MODELING OF WIRELESS CHANNELS	3
1.3.	RESEARCH GOALS	3
1.4.	CONTENTS	4
2.	DISCRETE MODELING OF THE WIRELESS CHANNEL	6
2.1.	CHARACTERIZING THE SIGNAL TO NOISE RATIO VARIATIONS	6
2.1.1.	The Finite State Markov Channel Model	8
2.1.2.	Hidden Markov Models	12
2.1.2.1.	Characterization of HMMs	13
2.2.	CHARACTERIZATION OF ERROR SOURCES	15
2.2.1.	The Finite State Channel	15
2.2.2.	Source modeling	16
2.2.3.	Error source modeling	16
2.2.4.	Problems related to hidden Markov models	17
2.3.	THE IEEE 802.11 CHANNELS	18
2.3.1.	Detection of frame losses in 802.11 channels	19
2.3.2.	Rate variations in 802.11 channels	20
2.3.3.	Effects of frame losses on upper layers	20
3.	EXPERIMENTAL DESIGN AND METHODOLOGY	22
3.1.	NOTATION OF FRAME LOSSES	22
3.2.	EXPERIMENTAL DATA ACQUISITION	25
3.2.1.	Experimental sites and statistics	25
3.2.2.	Setup details for characterizing frame losses	28
3.2.2.1.	Frame loss trace specifications	29
3.2.3.	Setup details for characterizing rate variations	30
3.2.3.1.	Rate variations trace specifications	31
3.2.4.	Limitations imposed by the MAC and PHY layers	31
3.2.5.	Experimental Design	32
3.2.5.1.	Frame loss experiment design	32
3.2.5.2.	Rate variations experiment design	35
3.3.	CONSTRUCTION OF MARKOVIAN MODELS FOR FRAME LOSSES	37
3.3.1.	FSMC model construction	37
3.3.1.1.	Model construction based on the characteristics of the fading envelope	37
3.3.1.2.	Model construction based on experimental data	40
3.3.1.3.	Summary of FSMC construction	41
3.3.2.	Hidden Markov model construction	42
3.3.2.1.	HMM construction based on experimental data	42
3.3.3.	Summary of models constructed for characterizing frame losses	45
3.4.	CONSTRUCTION OF MARKOVIAN MODELS FOR TRANSMISSION RATE VARIATIONS	45
3.4.1.	FSM construction	46
3.4.2.	HMM construction	47
3.4.3.	Summary of models constructed for characterizing rate variations	48
3.5.	COMPOSITE MODEL CONSTRUCTION	49
3.6.	VALIDATION OF FRAME LOSS MODELS	50
3.6.1.	Validation of the frame loss FSMC model output	50
3.6.1.1.	The Kolmogorov-Smirnov test for two independent samples	50
3.6.1.2.	Validation of the FSMC model output against experimental data	51

3.6.2.	Validation of the assumption behind the hidden Markov model construction.....	52
3.6.2.1.	Contingency table construction for validating the Markovian assumption.....	52
3.6.3.	Validation of the hidden Markov model frame loss output.....	54
3.6.4.	Validation of frame loss processes by analyzing the effects on upper layers.....	55
3.7.	VALIDATION OF THE MODEL FOR THE TRANSMISSION RATE VARIATIONS.....	55
3.8.	VALIDATION OF THE COMPOSITE MODEL.....	56
3.9.	VALIDATION SUMMARY.....	56
3.10.	SIMULATION PLATFORM AND GENERAL IMPLEMENTATION FEATURES.....	56
3.10.1.	General simulation setup in Modeler.....	58
4.	EXPERIMENTAL OBSERVATIONS AND ANALYSIS.....	60
4.1.	EXPERIMENTAL RESULTS OF FRAME LOSS PROCESS.....	60
4.1.1.	Frame loss distributions observed with 802.11b.....	61
4.1.1.1.	Office environment.....	61
4.1.1.2.	Residential Environment.....	63
4.1.2.	Frame loss distributions observed with 802.11a.....	66
4.1.2.1.	Office environment.....	66
4.1.3.	Summary of results for 802.11b and 802.11a systems.....	70
4.1.4.	Average duration of states for 802.11b and 802.11a systems.....	71
4.1.5.	2 ^k FACTORIAL DESIGN ANALYSIS OF THE EXPERIMENTAL RESULTS.....	76
4.2.	EXPERIMENTAL TRANSMISSION RATE VARIATIONS RESULTS.....	78
4.2.1.	Rate variations observed with 802.11b technologies.....	78
4.2.2.	Rate variations observed with 802.11a technologies.....	78
4.2.3.	2 ^k FACTORIAL DESIGN ANALYSIS OF THE EXPERIMENTAL RESULTS.....	81
5.	LIMITATIONS OF EXISTING MODELS.....	83
5.1.	LIMITATIONS IN THE CHARACTERIZATION OF THE MODELS.....	83
5.1.1.	Effects of parameter selection in Markov models.....	84
6.	MODELING OF THE CHANNEL AND MODEL VALIDATION.....	90
6.1.	MODEL RESULTS AND VALIDATION OF THE FRAME LOSS PROCESSES.....	90
6.1.1.	FSMC model.....	90
6.1.1.1.	Findings from the characterization based on the attributes of the fading envelope.....	90
6.1.1.2.	Findings from the characterization based on experimental data.....	93
6.1.2.	HMM.....	95
6.1.2.1.	Findings from the characterization based on experimental data.....	95
6.1.3.	Findings from the validation by looking at the effects on upper layers.....	101
6.1.4.	Summary of findings and frame loss processes modeling guidelines.....	102
6.1.4.1.	Model construction guidelines for frame loss processes.....	102
6.2.	MODEL RESULTS AND VALIDATION OF THE TRANSMISSION RATE VARIATIONS.....	107
6.2.1.	HMM and Markov model results for the representation of the transmission rate variations.....	107
6.2.2.	Findings from the characterization of rate variations with a Markov model.....	109
6.2.3.	Summary of findings and transmission rate variations modeling guidelines.....	110
6.2.3.1.	Model construction guidelines for rate variations.....	111
7.	CONCLUSIONS.....	114
APPENDIX A	ISSUES RELATED TO THE FSMC CHARACTERIZATION.....	117
APPENDIX B	VALIDITY AND ACCURACY OF THE FSMC.....	125
APPENDIX C	FINITE STATE CHANNEL NOTATION.....	130
APPENDIX D	PROBLEMS RELATED TO HMM.....	133
APPENDIX E	FURTHER SIMULATION DETAILS.....	139
APPENDIX F	INTERARRIVAL TIMES BETWEEN FRAMES.....	143
APPENDIX G	RESIDENTIAL ENVIRONMENT DATA.....	144
BIBLIOGRAPHY	145

LIST OF TABLES

Table 1 Necessary elements needed to describe the finite state Markov channel (FSMC)	10
Table 2 Example of a count of error sequences.....	24
Table 3 Criteria selection of experimental sites.....	26
Table 4 Experimental traces contents.....	29
Table 5 Example of an experimental trace.....	30
Table 6 Response variables details.....	33
Table 7 Factors details.....	35
Table 8 Response variables details for rate variations.....	36
Table 9 Factors details for rate variations.....	37
Table 10 Summary for characterization of the FSMC model for frame losses.....	42
Table 11 Summary of models for characterizing frame losses.....	45
Table 12 Summary of models for characterizing rate variations.....	48
Table 13 Two-way contingency table for the RSSI and a previous state sequence length of one.....	52
Table 14 Sample Two-way Contingency table for the RSSI and a previous state sequence length of two.....	54
Table 15 Summary of validation techniques.....	56
Table 16 Summary of results for 802.11b and 802.11a systems.....	71
Table 17 2^k design results for 802.11b experimental data for frame losses.....	77
Table 18 2^k design results for 802.11a experimental data.....	77
Table 19 2^k design results for 802.11b and 802.11a for the transmission rate variations.....	82
Table 20 Limitations in the characterization of traditional models.....	84
Table 21 Configuration settings for a four state FSMC.....	85
Table 22 Comparison of the simulation results vs. experiments for a two-state FSMC.....	94
Table 23 Sample K-S test results for the comparison of the states' distributions.....	95
Table 24 K-S test results for the comparison between HMM and experiments for 802.11b systems.....	99
Table 25 K-S test results for the comparison between HMM and experiments for 802.11a systems.....	100
Table 26 Comparison of the average transfer time of a file using FTP and several frame loss processes.....	101
Table 27 Two-state frame loss Markov model characterization parameters for 802.11b channels.....	104
Table 28 Two-state frame loss Markov model characterization parameters for 802.11a channels.....	104
Table 29 HMM frame loss characterization parameters for 802.11b channels.....	106
Table 30 HMM frame loss characterization parameters for 802.11a channels.....	106
Table 31 Two-sample K-S test results for the analysis of Markov models in transmission rate variations.....	110
Table 32 Four-state rate variations Markov model characterization parameters for 802.11b channels.....	112
Table 33 Eight-state rate variations Markov model characterization parameters for 802.11a channels.....	113
Table 34 χ^2 test for an 8, 16 and 32 state FSMC.....	122
Table 35 Average interarrival times (in seconds) for various frame sizes.....	143
Table 36 Average error and error free run duration (in seconds) at residential locations p and q.....	144

LIST OF FIGURES

Figure 1 Fading manifestations and degradations.	6
Figure 2 Partitioning the received SNR and assigning each interval to a state of the FSMC.	9
Figure 3 The finite state Markov channel model representation.	9
Figure 4 Variations of received SNR in time and frequency – frequency selective fading	12
Figure 5 Block diagram illustrating the training process of a HMM.	18
Figure 6 Characterization of frame losses.	23
Figure 7 Frame loss sequence.	23
Figure 8 Experimental results obtained at a SNR of 34 dB – 802.11b.	24
Figure 9 Physical setup where measurements were collected. Office (left), residential (right).	26
Figure 10 Generalized setup for capturing the frame loss traces.	28
Figure 11 Actual setup for capturing the frame loss traces.	30
Figure 12 Frame exchange detail for 802.11 systems.	31
Figure 13 The two-state Markov model with transition rates λ and μ	39
Figure 14 Normalized average duration of fades as a function of the fade depth.	39
Figure 15 Normalized positive crossing rate as a function of the fade depth.	40
Figure 16 Experimental data base computation of the good state duration for a two state Markov model	41
Figure 17 HMM construction of a first model for frame losses based on experimental traces.	44
Figure 18 Characterization of rate variations with a Markov model based on experimental data.	46
Figure 19 HMM construction of a first model for rate variations based on experimental traces.	47
Figure 20 Composite model construction.	49
Figure 21 Setup required to validate the FSMC model output.	51
Figure 22 A state of a Markov chain with its associated transition probabilities.	57
Figure 23 Simulation setup for comparing the frame loss traces effects on an FTP file transfer.	58
Figure 24 Presentation order of the experimentally observed distributions.	60
Figure 25 A sample of experimentally observed frame loss distributions at two different office locations.	62
Figure 26 A sample of experimental frame loss distributions at 1, 2, 5.5 and 11 Mbps at office location d.	64
Figure 27 Sample experimentally observed frame loss distributions for various frame sizes at office location d.	65
Figure 28 A sample of experimentally observed frame loss distributions at two different residential locations.	66
Figure 29 A sample of experimentally observed frame loss distributions at 2 different office locations.	68
Figure 30 A sample of experimentally observed frame loss distributions at 6 and 12 Mbps at location c.	69
Figure 31 Sample experimentally observed frame loss distributions for various frame sizes at office location c.	70
Figure 32 Experimentally observed average good and bad state durations for 1 Mbps transmissions.	72
Figure 33 Experimentally observed average good and bad state durations for 2 Mbps transmissions.	72
Figure 34 Experimentally observed average good and bad state durations for 5.5 Mbps transmissions.	73
Figure 35 Experimentally observed average good and bad state durations for 11 Mbps transmissions.	73
Figure 36 Experimentally observed average good and bad state durations for 6 Mbps transmissions.	74
Figure 37 Experimentally observed average good and bad state durations for 12 Mbps transmissions.	74
Figure 38 Experimentally observed average good and bad state durations for 24 Mbps transmissions.	75
Figure 39 Transmission rates variations observed at different office locations with 802.11b technologies.	79
Figure 40 Transmission rates variations observed at 3 different office locations with 802.11a technologies	80
Figure 41 SNR envelope partitioning and state assignment to a four state FSMC model.	85
Figure 42 Simulation setup for comparing the frame loss traces effects on an FTP file transfer.	86
Figure 43 Histograms from the simulation using the setup from Figure 42 with a 4 state FSMC.	87
Figure 44 Comparison of results for a four state FSMC.	88
Figure 45 Theoretical and experimentally observed average state duration ratios.	92
Figure 46 Error run and error free run distributions generated with a two-state Markov model.	93

Figure 47 A sample of experimentally and HMM histograms in the office environment at two locations.	97
Figure 48 A sample of experimentally and HMM histograms in the office environment.	98
Figure 49 Sample transmission rate variations evolution during 5000 frames	108
Figure 50 Sample transmission rate variations evolution during 5000 frames	109
Figure 51 Normalized fade durations vs. R/R_{avg} [dB].....	120
Figure 52 Autocorrelation functions R_R for the ISORA and FSMC models with 16 states at $f_D=0.001$	128
Figure 53 Forward algorithm next state representation diagram for the induction step.	134
Figure 54 Backward algorithm previous state representation diagram for the induction step.	135

1. INTRODUCTION

1.1. CURRENT STATE OF CHANNEL MODELING OF WIRELESS CHANNELS

In wireless communications the nature of the channel poses several challenges for data transmission. Wireless channels experience different phenomena than those observed in wired ones. In wired communications the transmitted signals do not experience all the degradations inherent in the wireless channel. Representing these signal degradations with models is not necessarily a straightforward process and a considerable amount of research has been performed to address this matter. [13] [23] [53]. It is desirable that models represent certain characteristics of interest with accuracy. Models can be constructed to represent the degradations or their effects at different levels. For example, it is not only important to understand the impact of the channel and the degradation on the signal itself, but on how this affects frames or packets as they are transmitted through the air. The importance of channel models lies on how they can be used to simplify the analysis, design and deployment of communication systems.

In performance studies of wireless communications systems it is important to make use of accurate channel models. For example, in simulation studies, a simple but accurate channel model is essential to explore diverse variables such as throughput or transfer time versus signal to noise ratio. Furthermore, channel modeling can also be helpful in understanding how to predict the behavior of the channel itself. The predicted results could then potentially be used to make decisions about the operation of a system, designing protocols for more efficient operation (e.g. improving energy efficiency), or for fine tuning the parameters of existing protocols. In this dissertation, the research interest lies on discrete channel models for emerging wireless LANs. These models are useful to characterize how bits or packets are lost when transmitted over wireless channels as opposed to how a signal is distorted, which is often of interest in receiver design.

Generally the quality of a channel is described by the value of the received signal to noise ratio. In the past, modeling of wireless channels has used the envelope of this ratio to characterize models. Due to the nature of the wireless channel this envelope varies in time and experiences fading. Fading may result in considerable degradation of the received the signal and therefore should be taken into account in the models. Usually wireless channel models take fading into account, by partitioning its time

representation and assigning each partition to a state in the model. Then by generating transitions between several states the variations of the original envelope are considered.

Markovian models have been used to represent the faded envelope as well as the frame or packet losses that occur because of fading. These models are quite popular and have been used in a variety of performance studies. For example, Chaskar et al. [16] and Chiani et al. [17] studied the performance of a transport protocol (TCP) over wireless links using a two state Markov model. Labiod [28] studied the performance of error correcting codes over wireless links with the same type of model. In all of these cases, the channel model determined how frames were lost at the link layer. The performance of other communication protocols, such as ATM [25] [39], over wireless links has also been studied via simulations using two state Markov models. Hidden Markov models have also been suggested to characterize losses in fading channels [49], for example in [51] these models were proven to be accurate in characterizing data transmission in cellular systems where the carriers are narrowband (30-200 KHz wide).

Markovian models are used because of the simplicity of their characterization and implementation [49]. For instance, [31] illustrates how such a model can be successfully used to characterize channels in which errors occurs in bursts. As the authors indicate in [31] these models are usually mathematically tractable. In particular, characterizing these models requires defining a few scalar parameters and matrices that determine how often the channel oscillates between ‘good’ and ‘bad’ states. Implementation is usually done by incorporating a simple error-generating mechanism for each state.

Markovian channel models have also been used in channel prediction studies [15] [22]. In these studies the channel model was used as a basis for comparing how well the prediction technique follows the channel behavior. In this case it is even easier to visualize how important an adequate channel model is. This is because the output of a simulated channel was used to measure the accuracy of the prediction technique

Several validity studies for Markovian models have been performed in the past. These have focused their efforts in verifying the legitimacy of fundamental assumptions and the statistical characteristics of the model. For example, Wang and Chang [54] showed under what conditions the Markovian assumption of Rayleigh fading is adequate while Tan and Beaulieu [46] extended this idea and suggested that a better approach is to analyze the autocorrelation function of the model. However, in the past, validity studies have mostly used simulated data to verify the accuracy of the models. Further issues related to the modeling of wireless channels and a discussion on the limitations of existing models are presented next.

1.2. ISSUES RELATED TO CHANNEL MODELING OF WIRELESS CHANNELS

Markovian models that have been proposed in the past have assumed simplified conditions for their mathematical characterization. In particular, fairly simple modulation schemes like BPSK or DPSK [53] have been assumed in order to simplify the construction of the model. Furthermore, for the characterization of the models it has been assumed that the underlying communications channel is not frequency selective. Even though these assumptions are valid for some cellular systems with simple modulation schemes, there are circumstances under which it is not clear if they hold. For example, current wireless local area network devices implement complex modulation schemes and operate over frequency selective channels. In these cases it is not clear how to characterize Markovian models or even if they are adequate to represent frame losses.

All characterization methods and results currently available in the research literature have been based on *simulated* channel conditions. For example the authors of [52], [53] or [56] used simulated versions of the faded envelope to generate their results. In the past no characterization or validation of the loss processes at the frame level has been performed. Experimental validation has been limited to study the accuracy and limits of operation of the fundamental Markovian assumption like in [6].

Furthermore, in past studies several characterization issues have been left open to discussion. Among these, it was not clear how the mechanism used to partition the received envelope affects the model, how many states are necessary to adequately represent the channel, how the packet size affects the model or how to relate the characterization parameters to the average received signal to noise ratio.

In this dissertation the interest lies on channels that are currently used by widely available wireless local area network technologies. In particular the focus will be on IEEE 802.11b and 802.11a channels. These are frequency selective, are currently used in a wide range of physical environments and implement new features like variable transmission rates that impact the performance of higher layer protocols. None of these characteristics have been previously taken into account in any model and it is unclear what effect if any they could have on the frame loss process.

1.3. RESEARCH GOALS

Taking into consideration the challenges detailed in the previous section this dissertation has three main goals.

- Study the accuracy of finite state Markov models and their characterization methods when these are used to approximate frame losses in IEEE 802.11 indoor channels.

- Study and then generate models for the transmission rate variations in IEEE 802.11 indoor channels.
- Generate a set of guidelines that will allow an accurate and simple characterization of Markovian models for frame losses in IEEE 802.11 indoor channels.

These studies have not been previously performed and their results would contribute to the understanding of how appropriate it is to use traditional models and new ones under specific conditions such as different values of signal to noise ratio, frame sizes and transmission rates. All these variables are relevant to current deployment of wireless local area networks devices and applications.

A distinguishable advantage of the approach taken in this dissertation is that since it will be based on experimental data it will automatically take into account the effects of frequency selective fading. This would eliminate the need for unnecessarily increasing the mathematical complexity of the models. The goals are not to analytically relate the underlying physical phenomena to the impact on frame losses but consider this issue in an empirical manner, similar to other channel modeling approaches at the signal level [34].

1.4. CONTENTS

Chapter 2 presents a basic conceptual description of the fading process in wireless channels. The Finite State Markov Channel (FSMC) characteristics and limitations are briefly discussed. Source and error modeling are also introduced as a basis for presenting hidden Markov models. This chapter finishes by briefly describing how to fit hidden Markov models to experimental data, a description of issues relevant to this research and some peculiarities of the IEEE 802.11 standard.

Chapter 3 discusses the experimental trials necessary to acquire data to validate traditional models and create new ones. The chapter starts by defining the structure of the experimental data that can be collected and the notation that will be used through the remainder of the dissertation. This chapter also details the characteristics of the experimental sites used for the experiments as well as the limitations encountered during the process. The main subject, modeling, is elaborated next. Particulars on the construction of Markov and hidden Markov models are given to explain how they can be created to characterize frame loss processes and the variations in the transmission rate in 802.11 systems. This chapter finishes by presenting the quantitative methods used to validate the modeling results.

Chapter 4 presents a summary of the vast amount of experimental data collected at the measurement sites. The chapter gives a glimpse of the tendencies present in the statistical distributions of frame loss processes. Results for two different types of environments, office and residential, and IEEE

802.11b and 802.11a wireless local area network technologies are gradually presented with a step by step change of the influencing factors. The chapter concludes by analyzing the experimental results with a factorial design approach that quantitatively illustrates the importance of each factor.

A discussion on the limitations in the characterization of previous Markovian models is presented in Chapter 5. This chapter illustrates the difficulties and inaccuracies that the construction of a model faces when traditional methods are used to describe it. These inaccuracies were found to be quite significant as shown by the sample models constructed throughout the chapter.

All the results of modeling the frame loss process and rate variations of the channel are presented in Chapter 6. In this chapter, the output obtained from both Markov and hidden Markov models is analyzed with the validation methods elaborated upon in Chapter 3. An insight on how well each model represents reality is given along with the guidelines for constructing practical and accurate Markovian models for 802.11b and 802.11a channels.

In the final chapter a summary of the relevant findings is given along with a discussion of future research work that can be explored in the area. The document ends by including in the Appendix detailed information about the creation and mathematical characterization of frame loss models as well as supporting numerical data obtained during the collection of data and the construction of Markovian models.

2. DISCRETE MODELING OF THE WIRELESS CHANNEL

The main interest of this dissertation revolves around the concepts related to the modeling of the wireless fading channel for WLANs. This chapter will start by covering the basics of fading, which is a degradation of the signal, a phenomenon intrinsic to wireless communications. Markov modeling of fading will then be built upon this introductory foundation.

2.1. CHARACTERIZING THE SIGNAL TO NOISE RATIO VARIATIONS

In wireless communication systems signals may travel through multiple paths between a transmitter and a receiver. This effect is called multipath propagation. Due to the multiple paths, the receiver of a signal will observe variations of amplitude, phase and angle of arrival of the transmitted signal. These variations originate the phenomenon referred to as multipath fading. Two manifestations, large-scale and small-scale fading characterize these variations [42]. These manifestations generate specific degradations in the signals. Figure 1 presents the fading manifestations and its associated degradations [42].

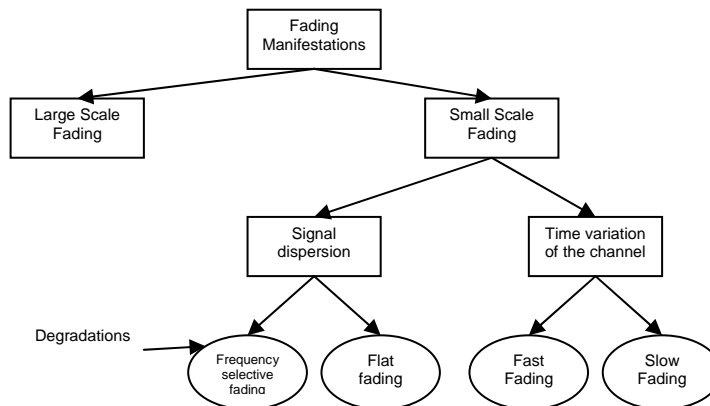


Figure 1 Fading manifestations and degradations.

The first fading manifestation, *large-scale fading*, refers to the path loss caused by the effects of the signal traveling over large areas. Large-scale fading characterizes losses due to considerably big physical objects in the signal's path like hills or forests. The path loss is characterized by a mean loss (due to the distance between the transmitter and the receiver and the propagation environment characteristics) and a variation around the mean loss.

On the other hand, *small-scale fading* characterizes the effects of small changes in the separation between a transmitter and a receiver. These changes can be caused by mobility of the transmitter, receiver or the intermediate objects in the path of the signal. Small scale changes result in considerable variations of signal amplitude and phase. Small-scale fading is also known as Rayleigh fading since the fluctuation of the signal envelope is Rayleigh distributed¹ when there is no predominant line of sight between the transmitter and receiver. When there is a predominant line of sight between the transmitter and receiver the fluctuations are statistically described by a Rician probability distribution function. Both large and small scale fading can be present in a wireless system.

Figure 1 shows two sub-manifestations of small scale fading. The first one, signal dispersion, refers to the time spreading of the signal. Dispersion causes the underlying digital pulses transmitted in the signal to spread in time. The second manifestation reflects the time variant behavior of the channel that is due to relative mobility between a transmitter and a receiver or the objects in the path of the signal. Both of these manifestations can be characterized in the time and frequency domain by fading degradation types.

In Figure 1 it is also shown that the degradation types of the dispersion manifestation are frequency selective and flat fading. From the time domain point of view, frequency selective fading occurs when the maximum spread in time of a symbol is greater than the duration of the symbol. Consequently, another name for this fading degradation is channel induced intersymbol interference. From the frequency domain point of view, frequency selective fading occurs when the spectral components of a signal are affected in different ways by the channel. In particular, frequency selective fading occurs when the channel's coherence bandwidth (the channel's bandwidth in which all components experience approximately the same fading characteristics) is smaller than the signal's bandwidth. When the conditions described above are not met (for frequency selective fading) the degradation is referred to as flat fading. In this case the channel characteristics are approximately flat for all frequencies.

The types of degradation for the time variation manifestation are fast and slow fading. From the time domain point of view, fast fading refers to the condition in which the channel's coherence time (the

¹ The Rayleigh probability distribution is: $p(r) = \frac{r}{b} e^{-\frac{r^2}{2b}}$ for $r \geq 0$ and $p(r)$ is zero for $r < 0$.

expected time duration during which the channel's response is invariant), is smaller than the symbol duration. Slow fading occurs when the coherence time is greater than the symbol duration.

From the frequency domain point of view, fast fading occurs when the signal bandwidth is less than the maximum frequency Doppler shift². Slow fading degradation occurs when the signal bandwidth is greater than the maximum frequency shift.

2.1.1. The Finite State Markov Channel Model

Wang and Moayeri [53] proposed the modeling of a Rayleigh fading narrowband channel using a Markov process with a finite number of states referred to as the Finite State Markov Channel (FSMC) model. The FSMC model originated as an extension of a simpler model proposed earlier, and known as the Gilbert-Elliott channel. In the FSMC, the fading process is related to the received signal to noise ratio (SNR). Such models are applicable *primarily to flat fading channels*.

The SNR is used since it is a common parameter that represents the quality of the channel [53]. For instance, the variations in the SNR can also affect the performance of other layers, like the link layer. At high average SNR the average number of lost frames due to transmission errors is expected to be low, the opposite occurs at low average SNR values. Therefore an accurate modeling of the received SNR can result in accurate channel models at the bit or frame level.

Figure 2 illustrates how the received SNR can be used in a FSMC model. First the SNR is partitioned into 'n' intervals or *levels*. Then each interval is associated with a state of a Markov process. The first interval starts at a level of zero SNR while the last one usually includes all received SNR values greater than a certain threshold. This procedure has been implemented in the past using a simulated received SNR that is *assumed to be valid for all frequencies and path losses*, conditions which are not valid for all wireless systems.

² The Doppler frequency, $f_D = v / \lambda$ characterizes the maximum frequency shift of the signals in a mobile environment. 'v' is the relative velocity between the transmitter and receiver and ' λ ' is the wavelength of the transmitted signal.

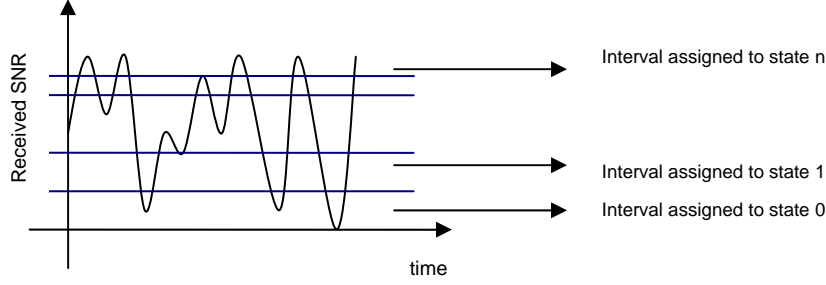


Figure 2 Partitioning the received SNR and assigning each interval to a state of the FSMC.

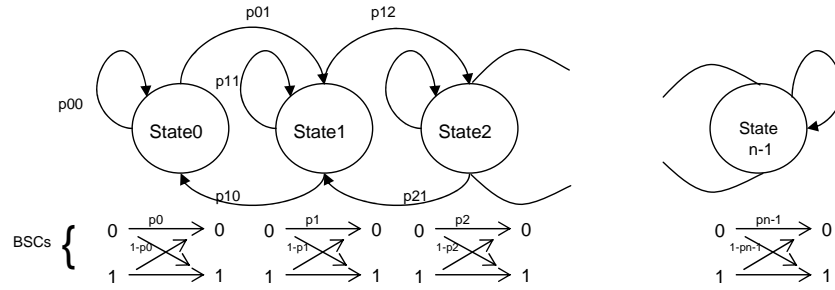


Figure 3 The finite state Markov channel model representation.

Figure 3 shows the FSMC represented by a chain of ‘n’ states. As seen in the figure, only transitions to the same state or to adjacent ones are allowed in the model. In the figure the in-state transition probabilities (p_{ii}) and the adjacent state transition probabilities (p_{ij}) are shown next to each arrow in Figure 3.

The goal of the model is to relate the varying nature of the channel with a loss process. For this, each of the ‘n’ states is associated with a different binary symmetric channel (BSC). The ‘n’ BSCs are shown in the lower part of Figure 3. In each state the associated BSC determines how a symbol being transmitted, for example a zero or a one, could be received in error. The individual probabilities of receiving a symbol in error are called crossover probabilities and are shown in the figure as $1-p_i$.

FSMC models are based on the theory of constant Markov processes³. Constant Markov processes have the property that the state transition probabilities are independent of the time at which they occur. These processes can be defined by a finite number of possible states that are usually represented

³ A Markov process with a discrete state space is also referred as a Markov chain [5, pg. 21]

by a set $S = \{s_0, s_1, \dots, s_{n-1}\}$ and a sequence of states $\{s_k\}$, $k=0,1,2,\dots$. Table 1 summarizes what is necessary to mathematically describe an 'n' state FSMC model [53].

The elements described in Table 1 must follow some constraints. Any element of the transition probability matrix \mathbf{P} should be between 0 and 1; the rows of \mathbf{P} should add up to one and the elements of \mathbf{e} should be between 0 and 0.5. Furthermore, not any set of \mathbf{P} , $\boldsymbol{\pi}$ and \mathbf{e} that satisfy these basic constraints will represent the physical channel [53]. Therefore it is necessary to establish other relationships between these elements and the channel characteristics, such as the modulation scheme, speed of the mobile and the frequency of the transmission. This would allow a correct characterization of the elements shown in Table 1. That procedure is discussed in greater detail in Appendix A.

Table 1 Necessary elements needed to describe the finite state Markov channel (FSMC)

Component	Notation	Description for an 'n' state FSMC
Transition Probability matrix	\mathbf{P}	A $n \times n$ matrix representing the probability of transition between states or into the same current state.
Steady state Probability vector	$\boldsymbol{\pi}$	A $1 \times n$ vector representing the steady state probability of being in any of the n states (additionally, $\boldsymbol{\pi} \mathbf{P} = \boldsymbol{\pi}$ and $\sum \boldsymbol{\pi} = 1$).
Crossover Probability vector	\mathbf{e}	A $1 \times n$ vector representing the different crossover probabilities of having a symbol in error in each of the n states.

Several issues arise during the characterization of the elements of Table 1. In particular it is important to understand how the partitioning scheme of the SNR, the number of states, the modulation and the coding affect the accuracy of the model.

Numerous partitioning schemes [52] [53] have been proposed to create the intervals that represent the states of the channel. Nonetheless no approaches have included in the partitioning criteria any fading characteristics. Only in [2] Aráuz and Krishnamurthy proposed a set of schemes that took into account the fading in the channel. That study compared several schemes and concluded that taking into account fading did not result in any major advantages in terms of the accuracy of the model parameters.

Appendix A discusses in detail the issues related to partitioning as well as a comparison between the existing schemes.

Just a few authors [6] [7] [8] have explored how the number of states affects the accuracy of a FSMC model. In all cases simulation studies have been used to generate the results; however no precise guidelines exist for the selection of the number of states the model should have.

The modulation and coding schemes of the model have been studied and in the past taken into consideration in great detail [53] [56] [57] [58]. It has been shown that a FSMC is accurate under a wide range of simple modulation and error correction schemes. Nevertheless no study has taken into consideration complex schemes such as those used in IEEE 802.11b and 802.11a wireless local area network technologies. Appendix A includes a detailed discussion of these issues as well as current relevant results.

The basic assumption made by all the FSMC model studies is that the underlying signal to noise ratio process follows the Markovian property. This property indicates that the probability of transition at a time 'n' to a new state only depends on the state at time 'n-1'. Extensive studies have been performed [46] [53] to understand the validity of this assumption. Those studies detail under what slow fading conditions the assumption is valid.

No experimental validation of the results for frequency selective channels has been performed in the past. Usually all validation studies compare the Markov modeling of fading or of frame losses with the results that are obtained by looking at mathematical models like the isotropic scattering, omnidirectional receiving antenna (ISORA) model [23][46]. In an ISORA model it is assumed that a signal that travels between a transmitter and a receiver experiences a spreading out (scattering) of its energy equally in all directions and is received using an omni directional antenna. Under these assumptions it is easier to obtain closed form expressions for some first-order statistics like the autocorrelation function of the envelope [46]. This result can then be used to compare a Markov model autocorrelation function to the ISORA one and quantify its differences. A comprehensive discussion of several validity studies is included in Appendix B.

Current literature **does not elaborate on the modeling of fading for frequency selective channels**. Under frequency selective conditions, how the signals are affected by the channel varies with frequency. Such a situation is illustrated in Figure 4 in which several SNR envelopes have different values that vary with time and frequency. Modeling frequency selective conditions, especially at the physical layer level, is not a simple process. This becomes even more intractable when the bandwidth of a signal is large and it spans frequencies that suffer *different* channel variations at the same time.

To overcome the effects of a frequency selective channel usually complex modulation schemes are used. Taking into account these schemes will further increase the difficulty of constructing a model

that relates the underlying phenomenon to the frame losses. In the next chapters it will be shown how it is not necessary to construct models that look at the signals physical characterization but only at its effects on frame losses in order to define an adequate frame level model.

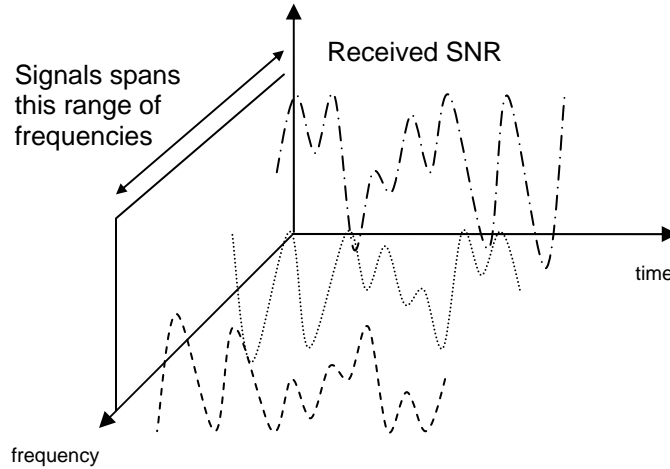


Figure 4 Variations of received SNR in time and frequency – frequency selective fading

2.1.2. Hidden Markov Models

As discussed in Appendix B, the application of first order FSMC is adequate under very slowly fading applications, that is, for short durations of time. Whenever there is a need to include the effect of very long channel memory the FSMC model is no longer appropriate. This is for example in the case of the study of fade duration distributions in fading channels [48]. Here, there is a need for Markov chains with larger memory, however since the number of states grows exponentially with the process memory, the approach is no longer practical [48]. In such cases other methods such as those that use hidden Markov models can be used.

Hidden Markov models (HMM) [37] are probabilistic functions of Markov chains (also known as Markov sources). These models can be used to study the fading process of a Rayleigh fading channel. We will first start by defining the general characteristics and concepts related to HMMs. Then we will proceed to describe how they are used to model fading.

A common discrete Markov process, like the one used in FSMC, is a stochastic process in which the outputs are observable. The outputs in this case are the set of states at each instant of time. Additionally, each state corresponds to some physical and observable event. These observable models

can be extended to include the case where the ‘observation is a probabilistic function of the state’. This results in a ‘doubly embedded stochastic process’ where one of the stochastic processes is not observable and hence the name hidden Markov model.

2.1.2.1. Characterization of HMMs

A HMM is characterized with the following elements.

- A set of the Markov chain states represented by $S = \{1, 2, \dots, n\}$. The number of states in the model is ‘ n ’. Even though these states are called ‘hidden’, in practical applications they are associated with some physical event.
- The set H of the observable output symbols in any state represented as $H = \{h_1, h_2, \dots, h_m\}$ with ‘ m ’ elements. ‘ m ’ is also called the alphabet size.
- The state transition probability distribution matrix $\mathbf{P} = \{p_{ij}\}$, where $p_{ij} = \Pr[\text{current state} = j \mid \text{previous state} = i] = \Pr[s_j \mid s_i]$
- The observed output symbol probability distribution matrices \mathbf{B} . \mathbf{B} are diagonal matrices whose elements b_j represent the probability $p\{h \mid s_j\}$ where $h \in H$ (if H is discrete).
- The initial state probability vector $\boldsymbol{\pi}$.

In this dissertation we do not intend to fully describe the characterization of HMMs, but to relate these models to fading processes. For this purpose, it is also necessary to partition the received SNR and assign states to the partitions. Once the model is established it is possible to compute the autocorrelation functions and other statistics of HMM [50]. Furthermore, it is possible to characterize error sources that can be used to generate block errors.

We are interested in describing the methods that can be used to fit a HMM to a specific fading or frame loss process. The first fitting method that can be used is the method of moments [50]. In this method the parameters of the model are found by equating the moments of the two models (i.e. HMM and ISORA for example or with the moments from experimental observations). This method has the problem that its system of equations is ‘ill posed’. This means that the moments are the same for very different models. Additionally, the selection of moments according to [50] is in general arbitrary. For example, finding a HMM with an autocorrelation function that resembles that of the fading process does not guarantee that the multidimensional probabilities associated with these processes are close. The method of the moments is generally used to obtain a first approximation that will be refined later with more advanced statistical methods.

A second fitting method consists in approximating multidimensional probability densities [50]. This method tries to answer the question of how to adjust the model parameters in order to maximize the probability of having a certain observable sequence. In more specific terms, if the observation sequence

O is given, such that $O = O_1, O_2, \dots, O_T$ ($O_i \in H$). How do we best describe it based on the model's parameters? This means we are trying to maximize the probability $p(O|\theta)$, where the model is $\theta = (\mathbf{P}, \mathbf{B}, \pi)$. The observation sequence used to compute the model parameters is called the training sequence.

There is no absolute optimal manner of estimating the model parameters to solve for the second fitting method. However, there are methods to locally maximize $p(O|\theta)$ using an iterative procedure. One of these iterative procedures is the Baum-Welch method (derived from the EM, expectation maximization method) [14]. Details of several procedures that can be applied to optimize the computational efficiency of the method are given in [50]. Additionally, an advantage of HMM modeling of fading processes is that it provides means to compute closed-form expressions for distributions of the fade duration and level crossing rates [50]. These expressions could be useful in the implementation of simulations.

Up to this point the actual HMM parameters have not been related to any real physical characteristic of the fading channel. In [40] and [50] it is illustrated how this is done. The channel is again connected to the HMM via the set of states S . As in the FSMC the fading amplitude needs to be quantized and an element of the set S is assigned to each quantization level. In these references there are no guidelines on how to select the threshold levels.

It is possible to compute the transition probability matrix \mathbf{P} via simulation of the fading envelope. This is done by partitioning the SNR and counting the total and the individual transitions between states that occur during a given period of time. The ratios of the latter quantities to the total transitions yield the desired quantities. In a similar manner the probability of the outputs of the model (\mathbf{B} matrices) can be computed by counting how many times a particular symbol h_i is generated in each state and dividing these values by the total number of output symbols observed in each state. This way, Turin and Van Nobelen [50] proceeded to compute the state duration distribution of a flat Rayleigh fading channel using the Baum-Welch algorithm. As shown in [50, Fig. 1] the approximation of the state distribution closely resembles that obtained from simulated data.

The advantage of using HMMs is that they provide enough flexibility to model different types of fading [48] [50]. Additionally if fading is modeled with a HMM then bit errors and block errors occurring over fading channels can also be modeled with HMMs [44] [45].

2.2. CHARACTERIZATION OF ERROR SOURCES

The FSMC model serves two main purposes. The first one is to approximate the received signal to noise ratio by discretely quantizing the envelope into states; the second, the generation of a loss process. In this section the characterization details of such loss processes is introduced.

A loss process in a FSMC is determined by the individual symbol error probabilities of the BSCs associated to each state. These probabilities are computed symbol by symbol based on the quantized SNR and the modulation scheme. As mentioned earlier, it is fairly easy to compute these probabilities with simple modulation schemes. However, when more complex schemes are used, like the ones present in spread spectrum systems and those used in wireless local area networks (802.11b and 802.11a technologies), this computation is not straightforward.

The symbol by symbol process created from the FSMC does offer the advantage that it could result in a good approximation if used to decide whether frames are received in error. This is because this decision can be made by looking at the number of errors that occurred and the error detection or correction schemes being used. On the other hand, a clear disadvantage of this approach is that the number of computations needed per frame in a simulation is quite high compared to an approach that does not look at all the individual symbols in a frame, but the entire frame as a whole.

The starting point for the characterization of loss processes will be a formal definition for the finite state channel developed by Shannon in 1948 [53]. When working with these models it is practical to use matrix notation instead of scalars. Appendix C details how this notation works by slowly migrating the explanation from basic probability expressions to matrix probability [48] notation.

2.2.1. The Finite State Channel

The finite state channel (FSC) allows the representation of a communications channel in which a given set of inputs when transmitted over the channel results in certain outputs. The channel itself can be in any state from a state space. Let S represent the state space and each of its elements the individual states so that $S = \{1, 2, \dots, u\}$. The transitions between states follow a state sequence also known as regime. If the set of input symbols is $A = \{a_1, a_2, \dots\}$ and the set of output symbols is $H = \{h_1, h_2, \dots\}$ then the probability of being in state s_{t-1} with a_t as input to the channel and going to state s_t with h_t as the output can be written as $\Pr(h_t, s_t | a_t, s_{t-1})$. In order to describe the FSC it is sufficient to count with the set:

$$\{S, A, B, \pi, P(h|a)\},$$

where:

S: channel state space

A: input alphabet

- B: output alphabet
- π : initial state probabilities vector
- $\mathbf{P}(\mathbf{h}|\mathbf{a})$: called the conditional matrix probability of the observing the output symbol $\mathbf{h} \in H$ given the input symbol $\mathbf{a} \in A$ (if the channel is not discrete the MP are replaced with the matrix probability distribution functions MPDF [48]).

Appendix C discusses in detail the notation and relations that can be established with these elements. It also illustrates how using matrix notation greatly simplifies the handling of them.

2.2.2. Source modeling

Source modeling can be used to model the channel input. The source can be modeled by an autonomous FSC (one in which the output does not depend on its input). A source can be described by the set $\{S_s, \pi_s, \mathbf{P}_s(\mathbf{a})\}$. Where S_s is the set of states, π_s the initial state probabilities vector and $\mathbf{P}_s(\mathbf{a})$ a matrix such that:

$$\begin{aligned} \mathbf{P}_s(\mathbf{a}) = \{\Pr[\mathbf{a}, j|i]\} &= \text{probability of transferring from state 'i' to 'j' and producing symbol } \mathbf{a}. \\ &= \Pr(j|i) \Pr(\mathbf{a} | i, j) \end{aligned}$$

This source is called a FINITE STATE GENERATOR (FSG) [48]. In a manner similar to that shown in Appendix C, using equation (7) one can compute the probability of observing a certain sequence of outputs $\mathbf{a}_1^t = \mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_t$ with the next expression:

$$\Pr(\mathbf{a}_1^t) = \pi_s \prod_{i=1}^t \mathbf{P}_s(\mathbf{a}_i) \mathbf{1}$$

As illustrated in Appendix C, the states constitute a Markov chain with transition probability matrix:

$$\mathbf{P} = [\Pr(j|i)] = \sum_{\mathbf{a}} \mathbf{P}_s(\mathbf{a})$$

Furthermore, if the probability of observing a symbol \mathbf{a}_t depends only on the current state, this is $\Pr(\mathbf{a}_t | i, j) = \Pr(\mathbf{a}_t | j)$ the model is called a *discrete hidden MARKOV model*.

2.2.3. Error source modeling

In error source modeling it is convenient to consider error sequences \mathbf{e}_1^t instead of channel outputs \mathbf{h}_1^t since the errors are just deterministic functions of the channel inputs and outputs. Appendix C develops expressions used to handle the elements of finite state sources and error sources.

The probability of observing a sequence of errors $\mathbf{e}_1^t = e_1, e_2, \dots, e_t$ given a sequence of inputs \mathbf{a}_1^t can be computed as:

$$\Pr(\mathbf{e}_1^t | \mathbf{a}_1^t) = \pi \prod_{k=1}^t \mathbf{P}(\mathbf{e}_k | \mathbf{a}_k) \mathbf{1}$$

If the matrices of the form $\mathbf{P}(\mathbf{e}|\mathbf{a}) = \mathbf{P}(\mathbf{e})$ that is, they do not depend on the input \mathbf{a} , then the channel is called symmetric. With this:

$$\Pr(\mathbf{e}_1^t) = \pi \prod_{k=1}^t \mathbf{P}(\mathbf{e}_k) \mathbf{1}$$

One can use a HMM to model the sequence of states with the set $\{S, E, \pi, \mathbf{P}(\mathbf{e})\}$, where E is the set of possible errors.

The symbol error probability can be expressed as:

$$\Pr(\mathbf{e}) = \pi \mathbf{P}(\mathbf{e}) \mathbf{1}$$

2.2.4. Problems related to hidden Markov models

There are three basic problems that are inherent to HMM characterization. Only one of these problems is relevant to this dissertation. Nevertheless since all problems are related, a description of all of them is included in Appendix D. By solving these problems it is possible to characterize a HMM based on experimental data in an optimal manner.

The first problem refers to computing the probability of observing a sequence of output symbols given a specific model. In particular if the observation sequence of output symbols is $O = O_1, O_2, \dots, O_T$ (where $O_i \in \text{set of output symbols}$), the goal is to efficiently compute $P(O|\text{model})$. One approach would be to compute $P(O|\text{model})$ by enumerating every possible sequence of states of length T, calculate the probability of occurrence of each sequence and then adding up the results to obtain the joint probability over all state sequences. This procedure is inefficient since it involves a considerable number of multiplications and additions. To solve this problem a procedure called **forward algorithm** can be used; this computes the probability of observing partial sequences of O, considerably reducing the number of operations needed.

The second problem is not of interest for this document. In this problem the goal is to compute the probability of observing a certain sequence of states given a sequence of observed output symbols. To solve this problem the Viterbi algorithm can be used.

The third problem determines values for the model parameters such that the probability of observing a given sequence of output symbols is maximized. There is no analytical solution to the problem of generally maximizing the probability, but locally maximized probabilities can be found by

iterative procedures such as the Baum-Welch method. This method will be used in later chapters to obtain HMMs that represent frame losses.

The Baum-Welch method operates by using a first approximation of the solution, labeled θ in the next figure. This first approximation is generated by taking the experimental data and computing the elements π , \mathbf{P} , \mathbf{B} by counting the occurrences of total and individual transitions and output symbols for each state. After θ is computed a second approximation of the model, labeled $\bar{\theta}$, is computed using the re-estimation equations shown in Appendix D.

The second approximation is used again to re-estimate the model with the re-estimation equations. Then the procedure is repeated iteratively until the differences between the new parameters and the old ones are not significant. In particular, for obtaining the results in later chapters the procedure was stopped when the values of the transition matrix probability \mathbf{P} changed less than 10^{-4} between iterations. This limiting point where the algorithm stops is not part of the re-estimation procedure or the description of the algorithm. The next figure illustrates the procedure.

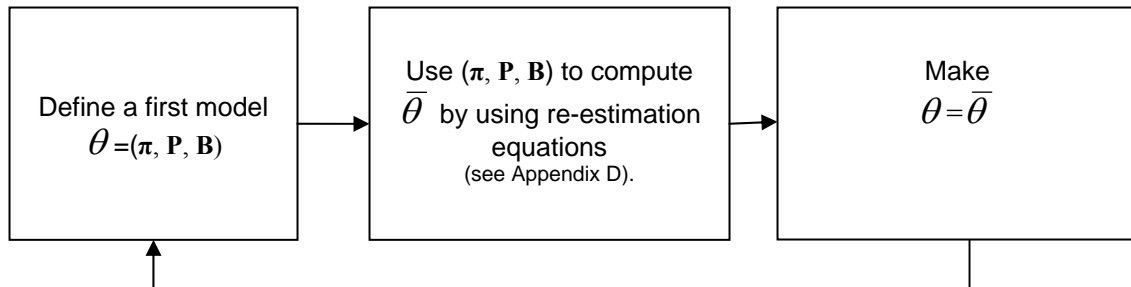


Figure 5 Block diagram illustrating the training process of a HMM.

2.3. THE IEEE 802.11 CHANNELS

The IEEE 802.11 standard, published in 1997, details the characteristics of the medium access and physical layers for what currently is a widely accepted access technology for wireless local area networks. The standard specifies how communications at 1 and 2 Mbps should be implemented in a 2.4 GHz band (ISM unlicensed band). The 1997 standard was followed by a 1999 supplement denominated 802.11b

which specifies operation at 5.5 and 11 Mbps in the same band. In late 1999 a high speed physical layer specification supplement, denominated IEEE 802.11a, was approved. This high speed specification details the communications mechanisms to provide transmission rates of 6, 9, 12, 18, 24, 36, 48 and 54 Mbps in a 5 GHz band (U-NII unlicensed band). Currently the last addition to the original standard is the IEEE 802.11g amendment approved in mid 2003. This amendment details how “further higher data rate extensions” of 22 and 33Mbps should be implemented in a 2.4 GHz band while maintaining backwards compatibility with 802.11 and IEEE 802.11b devices.

At the physical layer level, the 802.11 and 802.11b standards specify three kinds of physical layer “units”. The first one using frequency hopping spread spectrum technology and the second one using direct sequence spread spectrum (DSSS) technology. The DSSS unit, which is of interest in this document, can operate in any of 11 channels in a 2.4 GHz band (for North America). The third unit defined in the standard specifies operation in the infrared region of the spectrum. On the other hand, the 802.11a supplement specifies the operation of a radio unit using orthogonal frequency division multiplexing (OFDM) in any of 12 channels in a 5 GHz band.

At the MAC layer level the 802.11 standard defines three services. The first service, referred to as an asynchronous data service allows peer link layers to exchange MAC service data units (MSDU) on a best effort basis. The second service provides the adequate security means to authenticate and encrypt information transfer between two stations. The last MAC service is a MSDU ordering service that may reorder broadcast and multicast units in relation to unicast data units.

In this dissertation we will use the notation “802.11b” for those channels that can operate at 1, 2, 5.5 or 11 Mbps. To refer to those channels that operate at 6, 9, 12, 18, 24, 36, 48 or 54 Mbps we will use the “802.11a” notation. The notation “802.11” will be used to refer to both the 802.11b and 802.11a technologies.

2.3.1. Detection of frame losses in 802.11 channels

The 802.11 standard details the operation of devices at both the physical and link layers. The physical layer is further subdivided into two layers, a physical medium dependent (PMD) sublayer and a physical layer convergence procedure (PLCP) sublayer. The PMD sublayer defines the actual transmission characteristics that should be used by two stations transmitting information over the wireless medium. The PLCP sublayer which maps the MAC protocol data units (MPDU) to a framing scheme suitable for the PMD sublayer.

Before passing MPDUs to the PMD sublayer the PLCP sublayer adds an additional preamble and header. The preamble includes synchronization and frame delimiters bits. The header among other thing

indicates the transmission speed that should be used for transmission and reception of the MPDU. The header also includes a CRC-16 field that covers all the other fields in the header.

Frames (which are also referred as MPDUs) at the MAC layer level are constructed with three components. The first one is a variable length header which includes control information such as duration and addressing. The frame body is the second component, its length is variable and carries diverse information depending on the frame type. The last component is a frame check sequence which implements a 32 bit CRC that is computed over the header and body and allows the detection of errors in received frames.

As explained in Chapter 1 the multipath propagation effect affects to a great extent the correct detection of transmitted symbols, furthermore noise at the receiver creates additional reception challenges. Frames can be discarded at a receiving station when either the CRC-16 in the PLCP header fails or when the CRC-32 in the MPDU fails. When frames are discarded by the receiving PLCP entity these are not passed or reported to upper layers so it is not possible to detect those losses without directly obtaining them from the receiving WLAN card. Since the firmware inside the manufacturers cards is proprietary and not open source those losses cannot be accounted for. However, when the PLCP delivers an MPDU to the MAC layer, it is stored in memory before the error detection takes place. The 32 bit CRC used will allow the detection of all single, double and most triple single bit errors. Therefore MAC layer information can be used to keep track of the statistics of error and error free received frames.

2.3.2. Rate variations in 802.11 channels

The IEEE 802.11 standard published in 1997 establishes that devices that conform to the standard can perform “dynamic rate switching with the objective of improving performance”. The algorithm to perform the switching is not specified in the standard. However, in broad terms, rate switching allows a station to vary its transmission rate based on the measured SNR. That way, when the SNR falls below a certain level a lower transmission rate with a corresponding modulation scheme is used. In a similar way when the SNR increases above a certain level an analogous operation takes place.

2.3.3. Effects of frame losses on upper layers

The channel characteristics may impede stations in an 802.11 WLAN from maintaining continuous communications among them. From an upper layer perspective, discontinuity in the correct reception of frames will usually hinder the correct interpretation of these losses. For example in the case of transport layer communications that use TCP, continuous losses of frames could be misinterpreted as congestion in the intermediate network.

In particular in the case of TCP, frame losses may generate timeouts in the active connections. This is because losses at the link layer will be recovered at this level by retransmissions that will delay the transmission of subsequent frames and therefore of subsequent TCP data units. When a TCP connection times out it will retransmit the data that has not been acknowledged. If the receiving station continues to receive frames in error (or if the station is temporarily disconnected) these successive TCP retransmissions will also be delayed. When the information originally transmitted and the retransmitted one finally reach the destination, the receiver will have to discard any duplicated information. In such a situation wireless bandwidth and battery power have been unnecessarily used by mobile stations. Furthermore, if TCP times out it slows down its transmission rate and enters a phase known as 'slow start'. During this phase a TCP sender will limit the amount of data units it places into the network believing there is congestion, even though there is none.


3. EXPERIMENTAL DESIGN AND METHODOLOGY

The theoretical basis for channel modeling was developed in chapter two. In order to compare the traditional models described there with experimental data and circumvent their limitations, this chapter presents the experiment design to collect data, construct models and validate them. We will first discuss how the frame loss process can be characterized. Then a description of the experimental sites and the devices used to collect the data follows along with a discussion of their limitations. The design response variables, factors and levels are then presented. The chapter then proceeds to discuss the construction of Markovian models for the frame loss process and for the transmission rate variations. The validation methods for these models are presented in the last section.

3.1. NOTATION OF FRAME LOSSES

The goal behind the characterization of frame loss processes is the construction of a model that represents the loss behavior of the channel. In general this means that such a model should indicate periods of time when frames are received in error or are error free. These periods of time should have similar characteristics to those observed in actual experiments.

One can characterize a frame loss process by taking into direct consideration the errors that are detected at a receiver's MAC level. The information reported by this layer allows the collection of traces that are sequences of frames received in error or error free. These traces are basically a series of zeros and ones that represent frames received either correctly or in error respectively.

To further understand how the traces can be collected and their structure, let's assume that frames of fixed length are being transferred at a constant rate between two 802.11 stations. The following figure illustrates such a situation. In the figure, the sending station wirelessly transmits frames to a receiver. Some of these frames can get corrupted during their transmission and arrive in error; these are marked with the  symbol in Figure 6.

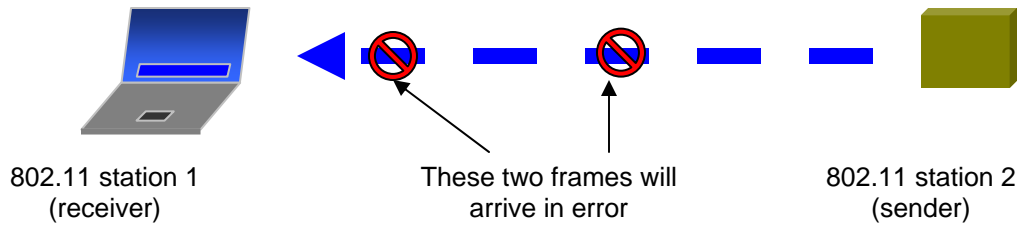


Figure 6 Characterization of frame losses.

In this example, the first and third frames transmitted by the sender arrive in error at the receiving station. In this particular case the observed frame trace sequence would be: 10100. This is because frames that arrive in error are represented by a one and frames that are error free are represented by a zero.

To understand how the duration of error and error free periods can be characterized, let's assume that a longer sequence of received frames is captured. For example:

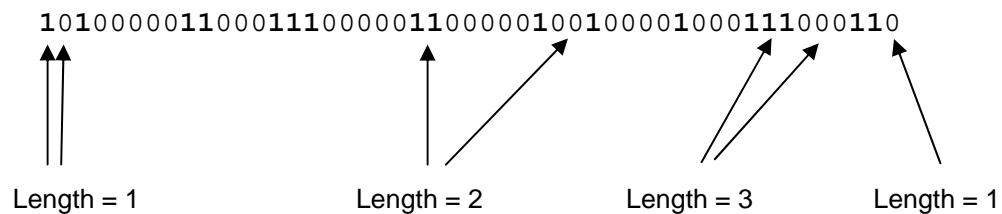


Figure 7 Frame loss sequence.

From the above sequence one can count that the number of *error* sequences of length one is five. This is because there five individual “1”s in the sequence. On the other hand, the number of *error free* sequences of length one is two. If one counts all the error sequences and error free sequences of different existing lengths the following table can be constructed.

Table 2 Example of a count of error sequences.

	Length of the sequence (in frames)				
	1	2	3	4	5
Count of <i>error</i> sequences	5	3	2	0	0
Count of <i>error free</i> sequences	2	1	3	1	3

By analyzing very long sequences it is then possible to construct a histogram of the count of error and error free sequences. An example of such a histogram is presented in Figure 8 for the case in which 1000-bytes frames are sent in an 802.11b system operating at 11 Mbps at an office location in which the average SNR at the receiver is 34 dB. These results were obtained by analyzing a sequence of 100,000 frames collected in an office environment.

Next to each of the histograms in the figure some basic statistics are also presented. These are the mean, maximum, minimum, standard deviation and number of samples. With all this information it is possible to characterize the frame loss process by using a model that generates sequences with distributions similar to those shown in Figure 8.

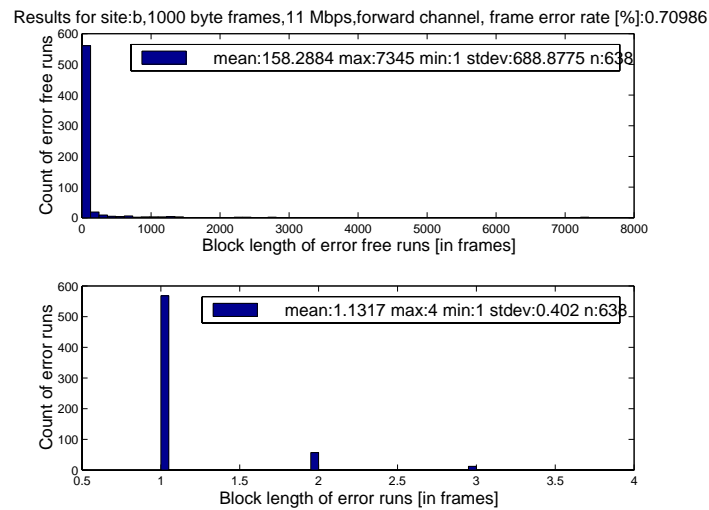


Figure 8 Experimental results obtained at a SNR of 34 dB – 802.11b.

Notice that the statistics in the figure are in “frame” units. The actual mean in seconds for the duration of the error and error free periods can be computed by multiplying the shown averages by the interarrival time between frames. The time between frames is a value that can be extracted from the experimental traces. This value varies according to the transmission rate, frame size and processing speed of the stations used during the collection of a trace. Appendix F includes a table with the observed values for each configuration; the values vary in the range of 12.6 ms to 0.6 ms. Since these values vary with the processing power of the stations they were only used in computations that involved the trace from which they were extracted from.

3.2. EXPERIMENTAL DATA ACQUISITION

In this dissertation, the characterization of frame losses with experimental data is employed for the construction of models such as Markov or hidden Markov models that represent the wireless channel. This section provides details on the sites selected for data acquisition, the experimental design itself, as well as the limitations encountered during the acquisition process.

3.2.1. Experimental sites and statistics

The sites selected to collect the experimental traces were chosen by first looking at typical current implementations in which IEEE 802.11b and 802.11a operate and then selecting environments similar to those that are common nowadays. Currently it is extremely popular to find 802.11 implementations in places such as offices, cafeterias, libraries, airports, plazas and residences. Collecting traces for every single type of environment can be done. However, we believe that looking at two typical cases, offices and residences, should provide enough valid information that can be used to understand how these channels behave at the frame level. This kind of approach is common in wireless channel modeling where models are developed for different environments [34].

The *office environment* used for the test was that from 4th floor of the School of Information Sciences, outside the Wireless Telecommunications laboratory. This environment is composed of concrete ceilings and floors and dry walls with metal framing. This type of setting resembles that found in typical offices.

The *residential environment* was composed of wooden floors and ceilings and dry walls with wooden frames. This setting also resembles the circumstances found in typical American apartments or houses. Figure 9 details the general distribution of the objects in both setups.

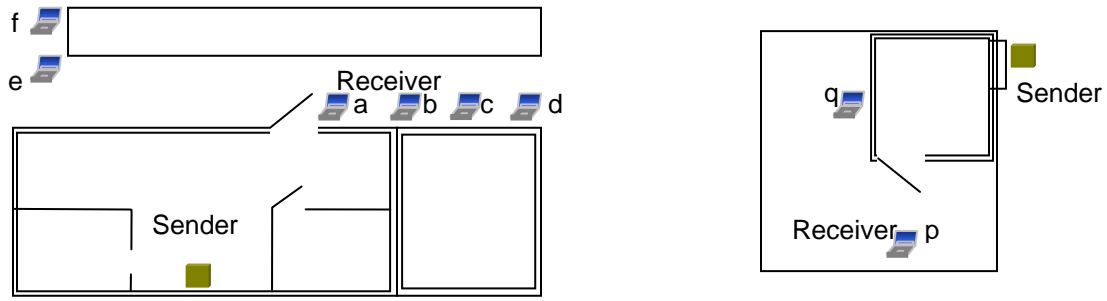


Figure 9 Physical setup where measurements were collected. Office (left), residential (right).

For both sites it was important to employ measurements that resemble the typical operating conditions in terms of distance and line of sight (LOS). Table 3 summarizes the relevant points considered for the selection of the measurement locations labeled **a** through **q** in the figure. As explained in the table the specific conditions selected for the experiments resemble those from today's implementations.

Table 3 Criteria selection of experimental sites

	Typical	Conditions selected for the Office (4 th floor SIS)	Conditions selected for the residential setting
Distance between transmitter and receiver	Vendors like Cisco and Proxim suggest maximum distances of 80 m. In an office or a residence you could see distances varying from a few meters to the maximum suggested.	Distances vary from 10 to 25 m.	Distances vary from 5 to 10 m.
LOS between transmitter and receiver	Because of partitions, walls and doors there is usually no LOS.	All measurement locations had no LOS.	All measurement locations had no LOS.

At both the office and residential sites the selection of the individual measurement locations (**a** through **q**) was done by first looking at the percentage of lost frames in each site. It was desirable to have locations that presented both low and high frame losses since this would allow the creation of models for a wide range of conditions. In particular at the office, locations **d**, **e**, and **f** were selected because at these places, even when the lowest transmission rates were used (1 Mbps for 802.11b and 6 Mbps for 802.11a)

the percentage of lost frames was above 50%. In the next chapter it will be illustrated how this high percentage of losses is not a typical operating condition in an office or a residence but a critical one. High frame loss situations such as those at locations **d**, **e** and **f** will provide a good understanding of the loss process under bad conditions.

In the office setting it was also necessary to include a corner in the path of the signal between the sender and locations **e** and **f**. The corner significantly reduced the average observed SNR, which resulted in higher frame losses. The same criterion was used to select the residential locations **p** and **q**. In the residential case since the distances were smaller it was necessary to lower the power of the transmitter to increase the percentage of lost frames.

Because of the physical dimensions of the office environment, it was possible to test larger distances between the transmitter and receiver. In particular, in the office environment locations **e** and **f** were selected to collect data at the lowest transmission rates of 1 and 2 Mbps. At these low rates the effects of the environment on the frame loss process were not noticeable in locations **a** through **d**. An extremely low number of frames was lost (less than 0.01%) in locations **a** through **d** when the transmission rate was set at 1 or 2 Mbps. In locations **e** and **f** much higher percentages were observed. Locations **e** and **f** were not used to collect data at 24, 12 or 6 Mbps since the average SNR at those locations was extremely low (less than 5dB) and all frames arrived in error.

The transmission power used at the office site was 50mW. Higher transmission powers could have been selected. However, at this particular value in the setting selected it was possible to include corners, doors and typical distances between the devices and still observe high losses.

In the residential environment the shorter distances between the devices made it difficult to observe high percentages of losses when the transmission power was set at 50mW. A lower transmission power, 5mW, was used for the measurements. The lower power allowed the recording of several traces with a wide range of frame loss percentages.

Some significant characteristics of the selected sites are summarized next.

- The construction materials (ceiling, divisions, etc...) are different at both sites; this should provide an insight about whether there is a significant difference between the office and residential environments.
- All the sites have at least one dry wall in the direct path between transmitter and receiver. This is similar to the typical configuration found in offices and residences.
- The distances from the transmitter to the closest points (**a** and **q**) are 10 and 5 meters respectively. This resembles the distances in an office or residential setting.

- The distances from the transmitter to the farthest points (**e** and **p**) are 25 and 10 meters respectively. This resembles the typical maximum distances in an office or residential setting.
- Locations **a**, **b**, **c** and **d** are located around one dry wall corner from the transmitter.
- Locations **e** and **f** are located around one concrete corner from the transmitter. This corner was selected in order to see considerably low average SNR at the receiver.

3.2.2. Setup details for characterizing frame losses

The frame loss process characterization is based on the statistical analysis of frames that consecutively arrive in error or are error free. Ideally this process should be independent of the MAC layer, software and hardware limitations. As explained later in this chapter when a frame arrives in error the transmitter has to time out before resending the frame. During this timeout, samples of the channel are not taken and the characterization could lose precision. Therefore, a slightly different setup to that simplification presented in Figure 9 was used.

In order to avoid seeing a high number of frames arriving in error at the receiver that would delay the transmission of future frames, in the actual setup the distance between the transmitter and receiver was fixed. A third device that operated in promiscuous mode was used to capture the traces. This third device captured traces at different distances from the transmitter. Figure 10 illustrates the configuration used.

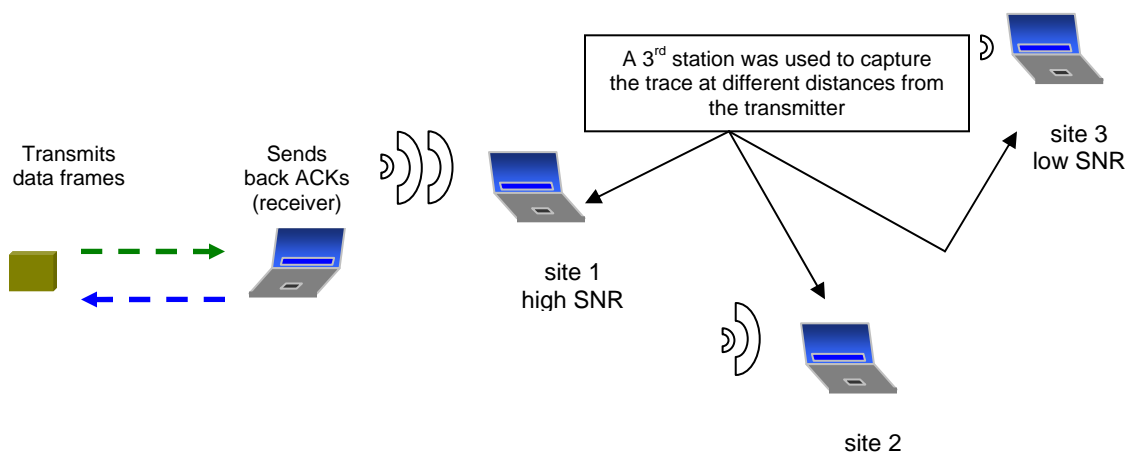


Figure 10 Generalized setup for capturing the frame loss traces.

Notice that the distance between the transmitter and receiver is small, less than one meter, fixed and with line of sight. Under these conditions a great percentage (close to 100%) of the frames that arrive at the receiver are error free and acknowledged. The capturing station can then be moved away from the transmitter to record the experimental traces. As the capturing station is moved farther away the mean SNR decreases and the percentage of data frames that arrive in error at the different locations (**a**, **b**, etc...) increases.

3.2.2.1. Frame loss trace specifications

For all the experiments the traces were captured using the WildPackets AiropeekNX v2.0 software tool running on a laptop with a Windows operation system. This tool allowed the capture of 802.11b and 802.11a traces that were later analyzed with statistics and math software packages. The actual captured traces contain the information described in Table 4.

Table 4 Experimental traces contents.

Column	Description	Comments
Frame number	Sequential number of captured frame	Around 100,000 frames were captured for each experiment.
Transmission rate	Transmission rate used to transmit the frame	For individual experiments the transmission of all data frames and acknowledgments was fixed to the set of available values.
RSSI	Received Signal Strength Indicator	Represents a value between 1 and 100% proportional to the average SNR at the receiver and captured during the reception of every frame.
Frame size	Frame size in bytes	Several payloads were selected in order to construct frames of 100, 500, 1000 and 1500 bytes
Flag	Indicates if the received frame arrived in error or error free	Only CRC frame losses are detected by the software tool used
Timestamp	Absolute time at which a frame was completely received.	-
Protocol	Protocol that the payload or control frame corresponds to	-

The version of the software tool used does not report the received SNR of every frame but the average RSSI which is proportional to it. The exact procedure of how the RSSI is computed by the hardware and software tools has not been disclosed by the vendors. A sample of an actual trace is shown in Table 5.

Table 5 Example of an experimental trace.

Frame Sequence	Tx Rate (Mbps)	RSSI	Flag (# means in error)	Frame size (bytes)	Timestamp (seconds)	Protocol
1	5.5	74%		1500	0	UDP
2	5.5	44%		14	0.000144	802.11 Ack
3	5.5	70%		1500	0.002681	UDP
4	5.5	42%		14	0.002815	802.11 Ack
5	5.5	74%		1500	0.005355	UDP
6	5.5	42%		14	0.005485	802.11 Ack
7	5.5	74%	#	1500	0.008144	UDP

The last table contains the first seven entries of a trace captured at a 5.5 Mbps fixed rate with 1500 byte data frames. Whenever an error is detected at the capturing station a “#” symbol is inserted in the “Flag” column. By using the information of the “Flag” column it is possible to construct the sequence of frames in error/no error. In our case only the frames carrying UDP information are of interest, since these are the ones with sizes being manipulated by the software application developed to control the transfers. The information from the RSSI column will be useful in the creation of hidden Markov models.

3.2.3. Setup details for characterizing rate variations

Rate variations occur according to a process that monitors the link quality in a transmitting station. Specific details about this process are in general not disclosed by the hardware manufacturers. However, it is known that this process takes into account the observed SNR to determine if it is appropriate to change the transmission rate in order to decrease the number of frames received in error. Taking samples to observe how this process evolves is fairly simple. Figure 11 shows the setup used.

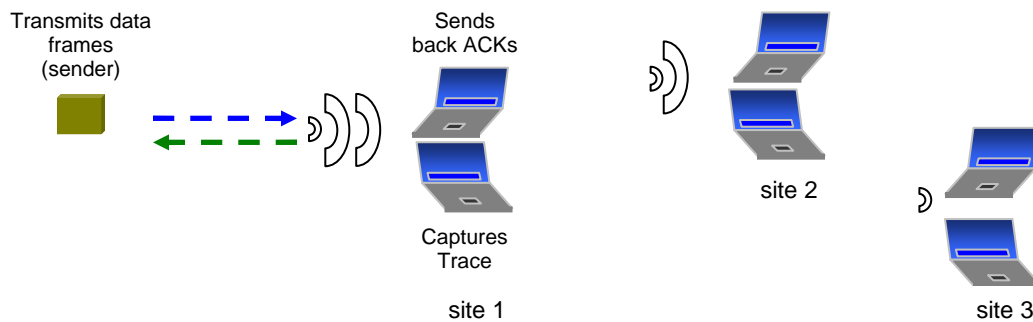


Figure 11 Actual setup for capturing the frame loss traces.

Two stations were used to capture the rate variations traces. One station was used to capture the trace while a second station was in charge of sending back acknowledgments to the sender. With this setup the amount of work that has to be done by each station decreases. With fewer tasks to perform the receiving station can reply to acknowledgements faster; this results in a more accurate trace.

3.2.3.1. Rate variations trace specifications

The rate variations traces were also captured using AiropEEK NX. The structure of these traces is the same as that presented in Table 5. For the rate variations analysis, the transmission rate column was used to obtain the information about consecutive runs at a given speed. In general these runs have a much longer duration than the error/no error runs, therefore it was necessary to collect longer traces to capture enough information. These traces included a total 500,000 frames.

3.2.4. Limitations imposed by the MAC and PHY layers

In an ideal approach the collection of frame losses in wireless channels should be independent of any limitations imposed by the measurement devices and the software running on them. In practice it is not possible to cancel all the effects of the elements that influence the data acquisition process. In terms of hardware and software, the frame loss traces are influenced by the 802.11 MAC and PLCP protocols and the processing speed of the stations involved in the measurements.

The 802.11 MAC protocol operates in a manner very similar to a stop-and-wait approach. This means that after a station places a data frame in the air, it waits until an acknowledgment is transmitted back by the receiving station before transmitting the next data frame. This basic exchange process is illustrated in Figure 12.

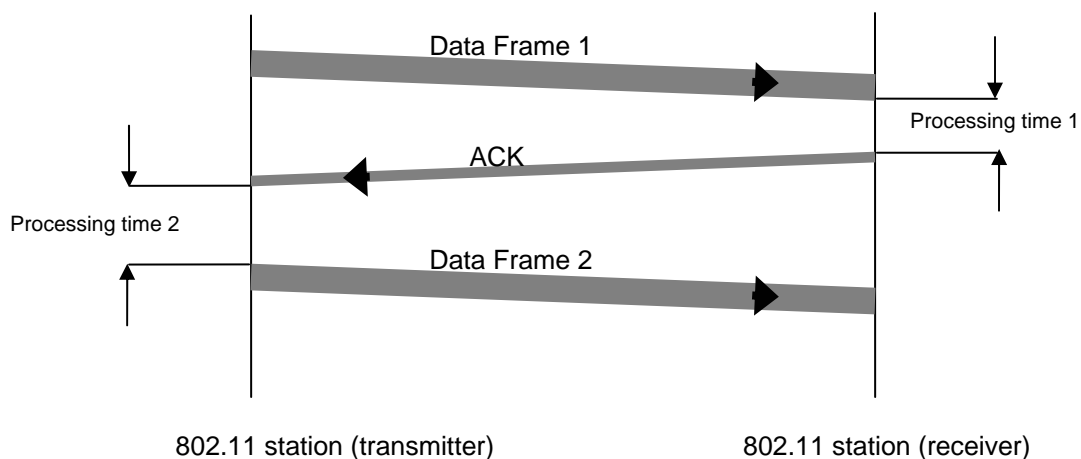


Figure 12 Frame exchange detail for 802.11 systems.

Notice that in the last figure, as soon as “*Data Frame 1*” arrives at the receiver it will be processed and an entry in the trace will be made (either a 1 or a 0). However during the processing time at the receiver (*Processing time 1*) and the transmission time of the acknowledgment it is not possible to send any more data frames. Furthermore, only after the acknowledgement is processed at the transmitter, “*Data Frame 2*” is sent. Meanwhile no measurements of the channel state can be made at the receiver. Without taking into consideration the propagation time of the signals, the period during which no measurements are taken can be expressed as:

$$\text{processing time 1} + \text{ACK transmission time} + \text{processing time 2} + \text{transmission time of “Data Frame 2”}$$

If we do not consider the propagation delay between stations, this period of no measurements has an average maximum value of 500 ms. This value was computed from a trace collected using 1,500 byte frames. The value varies with the frame size used in the transfers and with the processing power of stations.

The physical layer may also introduce an additional period of time during which it is not possible to measure the state of the channel. This occurs when an error is detected by the PLCP layer. However these errors are not reported to upper layers and therefore cannot be taken into consideration in the frame loss trace.

During the periods of time when no measurements are taken, it could be possible to have changes in the channel state. However the physical devices used do not offer means to circumvent this limitation. In the next chapters it will be seen how this limitation does not considerably affect the accuracy of the proposed models.

3.2.5. Experimental Design

With a basic setup, such as that presented in Figure 10, it was possible to design experiments that allow the collection of data for the characterization of frame loss processes and rate variations in 802.11 channels. Several experiments were designed for the collection of frame loss and rate variations data. The formal definition of these experiments is presented next.

3.2.5.1. Frame loss experiment design

Response variables.- These variables will help in the evaluation of the performance of the model. The first variable selected was the mean of the duration of the error or error free periods. This variable will be used in a preliminary comparison approach to understand the differences between the experimental data and the model. If the mean of these processes in the model do not match that of the

experiments it will be a sufficient measure to reject the frame loss process generated by the model. However if the means do match, this statistic is not sufficient to guarantee that the underlying distributions of the error and error free runs are the same.

To understand if the underlying distributions of each run generated by the model are similar to the experimental ones, histograms will be constructed. Further validation tests explained later in this chapter will be used to quantitatively understand the differences between the histograms of the models and the frame losses observed experimentally.

Higher order statistics could have also been selected as response variables. However the existence of simple statistical methods, like the two sample Kolgomorov-Smirnov test, that allow the comparison of the underlying distributions will be illustrated in later chapters to be a sufficient measure.

To understand how well the model fits the experimental data in terms of the effects at the transport layer, the transfer time of files between stations will also be considered as a response variable. This will be measured in a simulated environment that incorporates the file transfer protocol (FTP) over transport control protocol (TCP). This response is usually selected [16][17] to understand the effects of wireless channels on upper layers. The transfer times of 10 Mbytes files will be measured in a simulation that can use both the Markovian model and the experimental traces to discard frames.

Table 6 summarizes the variables chosen for the frame loss case and are valid for both 802.11b and 802.11a technologies. In the tables, the term “run” refers to periods of consecutive 1’s or 0s observed in the traces.

Table 6 Response variables details.

Response variables	Description	Response evaluation
Mean of runs	Computed for the length (in frames) of: 1. Error free runs 2. Error runs	Compare the results from the model with those from the experimental trace. A t-test will be used.
Histogram of runs	Computed for the distribution of: 1. Error free runs 2. Error runs	Compare the results of Kolmogorov-Smirnov tests for two independent samples to evaluate differences between the model and the experimental distributions.
Transfer time over a wireless link using a TCP/IP/WLAN 802.11b stack and FTP.	Computed for large file sizes (100 Mbytes) being transferred between two 802.11b stations.	Compare the results between a model driven and a trace driven simulations. A t-test will be used.

Factors.- Factors are elements that could affect the response. Each factor can be assigned several values referred to as levels. For IEEE 802.11b and 802.11a systems there are several factors that could affect the response. In particular it is possible to change the transmission rate, frame size, frequency, medium access method, power of the transmission between stations among other vendor specific factors. For the studies in this dissertation only the transmission rate, frame size and the power of the transmitting station were taken into consideration. The transmission power factor was not taken directly but analyzed through the average SNR at the receiver. The justification for the selection criteria follows.

Currently wireless local area networks are configured in way that the transmission rate between stations varies in time according to channel conditions, hence the need to include this factor in the experiments. All transmission rates available in 802.11b systems and in 802.11a systems were selected. .

The frame size is another element that is usually present in performance studies at the frame or packet level [16][17]. This is because the percentage of frames or packets lost varies with the frame size selected for transmissions. The levels selected for this factor varied from the most common level of 1500 bytes (corresponding to the maximum allowed in the standard) to 1000, 500 and 100 bytes. This wide range of levels would allow a better understanding of the behavior of the channel as a function of the frame size. The smallest frame size of 100 bytes (or in general small frames) did not generate useful data in all the experiments with 802.11a systems; the reason for this hardware barrier at the collecting station was not identified. Therefore small frame sizes were not used in 802.11a measurements.

The average SNR at the receiver was selected because it is a common parameter used to report channel quality and can be easily varied by selecting different locations for data collection. The values at the diverse locations were not selected directly but were measured when the locations were selected. As mentioned before the locations were selected in such a way that a wide range of percentage of frame losses was observed. Both low loss and high loss locations were needed since these conditions are usually observed in any typical installation in which users access the networks at different distances between transmitter and receivers.

The medium access method was not selected as a factor since current installations usually select by default the standard distributed reservation method for accessing the channel. Therefore this default access method was also used during the collection of the experimental data. The frequency of the transmissions was also left out as a factor with varying levels. To take into consideration the worst path loss case, the highest frequency was selected for the collection of the traces.

The following table summarizes the factors selected for the design.

Table 7 Factors details.

Factor	Levels	Description
Transmission rate	11, 5.5, 2 and 1 Mbps (for 802.11 b) 54, 48, 36, 24, 12 and 6 Mbps (for 802.11a)	All the allowable transmission rates were taken into consideration.
Frame size	100, 500, 1000 and 1500 bytes for 802.11b and 500, 1000 and 1500 bytes for 802.11a.	Different frame sizes were used to collect the data for the traces.
Mean SNR at the receiver	36, 32, 26 and 23 dB in an office and residential environments (for 802.11b) 17, 14 and 9 dB in an office and residential environments (for 802.11a)	For two different types of locations, office and residential, several distances between transmitter and receiver allowed taking measurements at different SNR values.

3.2.5.2. Rate variations experiment design

Response variables.- These are similar to those selected for the frame loss situation. In this case instead of taking into consideration the runs that describe the frame losses, the runs of periods of constant transmission rates were used. As before the mean of the runs for each rate are considered as a first measure of comparison.

If the mean duration of the runs differs significantly from the experimental ones the model will be rejected, however if they match it is necessary to perform further comparisons. In order to perform such comparisons the histograms of the duration of each transmission rate will be analyzed using a two sample Kolmogorov-Smirnov test. Table 8 summarizes the response variables selected for the experiments.

Table 8 Response variables details for rate variations.

Response variables	Description	Response evaluation
Mean of runs	Computed for the length (in frames) of: 1. Continuous runs at a fixed rate and analyzed for all the transmission rates	Compare the results from the model with those from the experimental trace.
Histogram of runs	Computed for the distribution of: 1. Continuous runs at a fixed rate and analyzed for all the transmission rates	Compare the results of Kolmogorov-Smirnov tests for two independent samples to evaluate differences between the model and experiment distributions.

Factors.- For the rate variations case, the frame size and the mean SNR at the receiver were selected as factors.

The frame size was selected as a factor because the percentage of frames lost varies with it. Therefore it is relevant to understand how the rate variations get affected by the choice of frame size. Frames sizes ranging from the maximum allowed, 1500 bytes, to 100 bytes (only for 802.11b) were incorporated in the measurements. These frame sizes were selected in order to have information of how the rate varies from the maximum allowable size to small frame sizes.

The other main factor that influences the transmission rate is the average SNR at the receiver. Stations closer to the transmitter are expected to use higher rates in comparison to those stations situated farther away. The same locations used to collect the frame loss process were selected since this will allow a characterization of both the rates and the loss process at a given average SNR value. Table 9 summarizes the factors and levels selected.

Table 9 Factors details for rate variations.

Factor	Levels	Description
Frame size	100, 500, 1000 and 1500 bytes for 802.11b and 500, 1000 and 1500 bytes for 802.11a.	Different frame sizes were used to collect the data for the traces.
Mean SNR at the receiver	36, 32, 26 and 23 dB in an office and residential environments (for 802.11b) 17, 14 and 9 dB in an office and residential environments (for 802.11a)	For two different type of locations, office and residential, several distances between transmitter and receiver allowed taking measurements at different SNR values.

3.3. CONSTRUCTION OF MARKOVIAN MODELS FOR FRAME LOSSES

Frame losses and rate variations can be modeled with similar methods if the distributions of consecutive runs are taken into account. Given a distribution, and its parameters, it may be possible to approximate experimental data characteristics by using Markov models. In particular, two types of models are of interest for the following discussion. These are the finite state and hidden Markov models. For both of these models, certain assumptions should be verified before the data is modeled. The validations of these assumptions and of the models themselves will be discussed in section 3.6.

3.3.1. FSMC model construction

The Finite State Markov Channel model can be used to describe the frame loss process occurring on the wireless medium. By definition, this model is a birth-death process in which transitions take place only between neighboring states. If two states are used, these neighboring states can directly represent periods of time in which all frames are received either in error or error free.

In this section the construction of the FSMC will be illustrated in two ways:

- Following the traditional method of constructing the model based on the characteristics of the fading envelope.
- Using experimental data obtained in a wireless local area network setting.

3.3.1.1. Model construction based on the characteristics of the fading envelope

This method has been widely used in the past to construct two-state Markov models, such as the one shown in Figure 13. The method assumes that a wireless signal with a given frequency is sent between two stations with no line of sight between them and that one of the stations or both are moving at a known velocity and direction.

The goal of the model is to assign durations to each of two states by using the curves from figures 1.3-4 and 1.3-5 developed by Jakes in [23]. These curves are reproduced via simulation in Figure 14 and Figure 15. The normalized average duration of a fade at a certain fade depth below the r.m.s value of the received envelope can be read from Figure 14. The normalized positive crossing rates in relation to the r.m.s value of the envelope are plotted in Figure 15. Both figures show results generated from the theoretical expressions [23] and simulations.

The next list illustrates the traditional procedure followed to assign durations to the bad and good states.

1. Assume that the relative speed between transmitter and receiver is v .
2. Assume that the center frequency of the carrier used in the transmission is f .
3. Compute the Doppler frequency as $f_D = v \times f$.
4. Select a fade depth value p . This value will be used to compute the duration of the bad state (s_I) of the model. By selecting this level one is assuming that whenever the received envelope is below p the channel is assumed to be in the bad state.
 - a. Let “ b ” be the mean normalized duration of a fade at level p . Obtain “ b ” by directly reading this value from Figure 14.
 - b. The mean duration of the bad state $\frac{1}{\mu}$ is equal to: $\frac{b}{f_D}$.
5. Let “ r ” be defined as the reciprocal of the sum of the duration of the good and bad states, this is:
$$r = \frac{1}{\text{duration}_{good} + \text{duration}_{bad}} = \frac{1}{\frac{1}{\lambda} + \frac{1}{\mu}}.$$
 - a. Let “ s ” be the normalized level crossing rates at fade depth p . “ s ” can be read directly from Figure 15.
 - b. Make $r = s \times f_D$.
 - c. The mean duration of the good state $\frac{1}{\lambda}$ is then equal to: $\frac{1}{r} - \frac{1}{\mu}$.

After computing the duration of both states it is possible to construct the model by assigning the mean durations to each state as the means of geometrically distributed periods of time. By definition the states of the Markov model have geometrically distributed duration.

With this approach, notice how there is *no way to directly relate the states’ durations to the actual average value of the received SNR*. This means that the signal variations are assumed to be the same for every value of the SNR. There are no clear guidelines on what the value of the fade depth “ p ” should be.

Furthermore there is no way of incorporating other factors such as the frame size or the transmission rate in the characterization. The model is assumed to be the same for all sizes and rates.

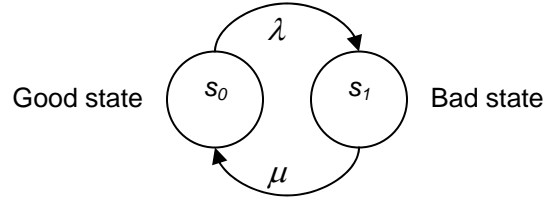


Figure 13 The two-state Markov model with transitioning rates λ and μ .

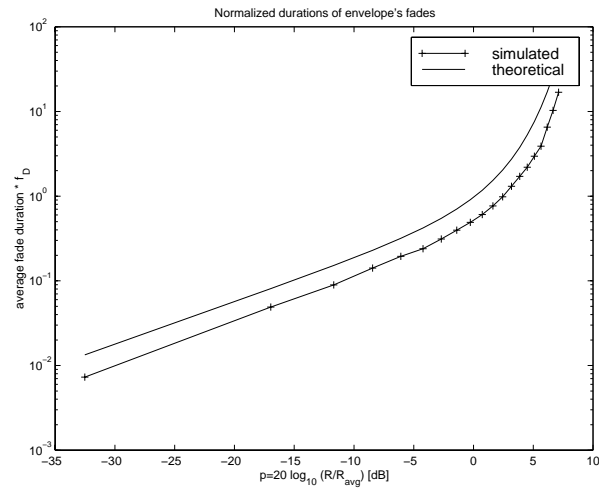


Figure 14 Normalized average duration of fades as a function of the fade depth

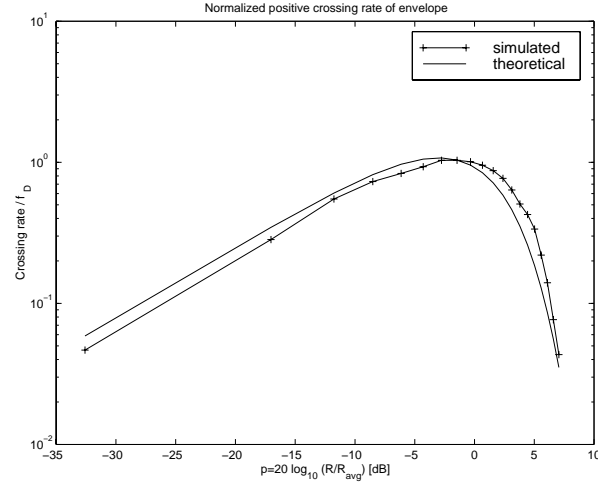


Figure 15 Normalized positive crossing rate as a function of the fade depth.

3.3.1.2. Model construction based on experimental data

The characterization based on experimental data can be performed by directly extracting the values for the states' durations ($1/\lambda$ and $1/\mu$) from the distributions of the error and error free runs in the experimental traces. The model can be then constructed by including these values in an event simulator, such as Opnet's Modeler, which includes advanced resources for the generation of geometrically distributed periods of time in finite state machines. Details on the computer simulations implemented can be found in Section 3.10 and in Appendix E.

By looking at the experimental results, which will be detailed in the next chapter, it was apparent that the mean duration of the error or error free runs observed in the experimental traces stayed constant. This was tested by taking each of the traces and computing their means and comparing them with a statistical t-test. The constant mean indicated that the underlying process can be assumed to be stationary. With this observation in mind the characterization was done as follows.

Figure 16 illustrates the characterization process for computing the state duration values. The figure shows the computation of the $\frac{1}{\lambda}$ value, the good state duration. A similar approach can be taken to compute $\frac{1}{\mu}$, the bad state duration.

The process consists in taking three error free runs and computing their means. Then taking an average of these three means and assigning the resulting value to $\frac{1}{\lambda}$. For the bad state it is necessary to

compute the means of experimental error runs and the average of their means to obtain the value to be assigned to $\frac{1}{\mu}$.

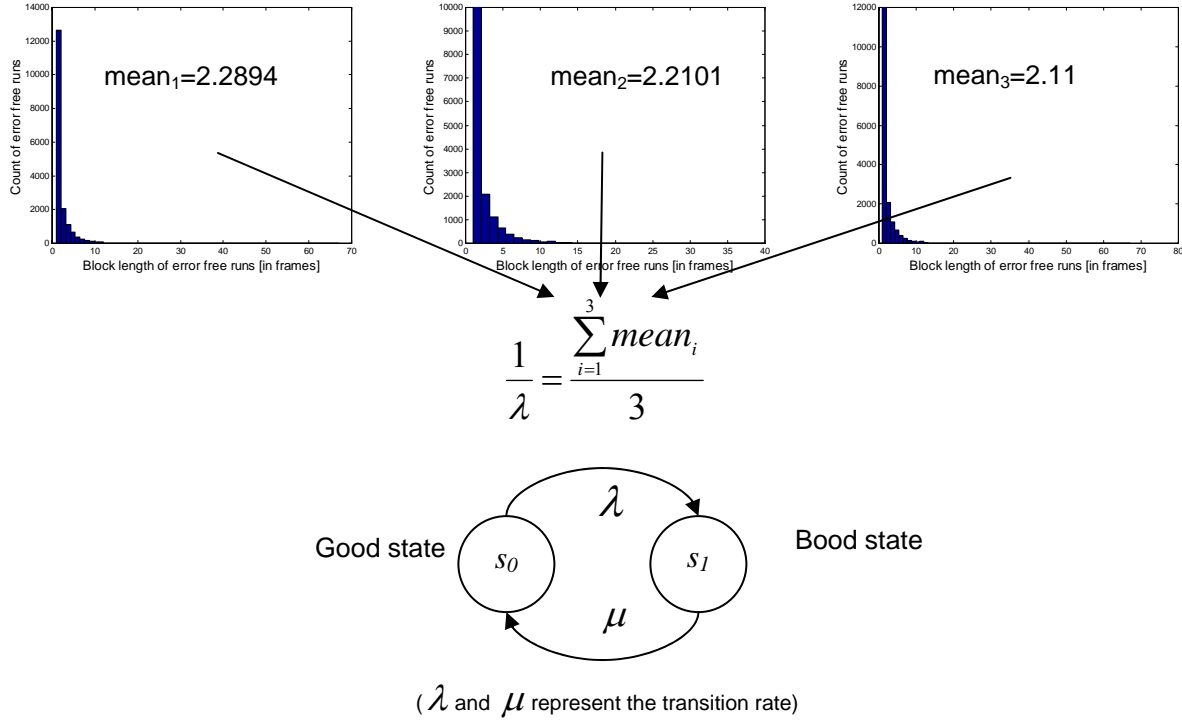


Figure 16 Experimental data base computation of the good state duration for a two state Markov model

After computing the durations of both states and in order to generate the frame loss process from the model, it is frequently assumed that all the frames arriving at the receiver during state s_0 (good state) will arrive error free, while all the frames during state s_1 (bad state) will arrive in error. It could also be possible to assign other percentages to the frame loss process in each state. However other percentages are not necessary to obtain an accurate model, this will be illustrated when the results are presented in Chapter 6.

3.3.1.3. Summary of FSMC construction

The next table summarizes the two methods presented for the characterization of a FSMC model for frame losses.

Table 10 Summary for characterization of the FSMC model for frame losses.

Method based on	Elements necessary for characterization	Can be obtained from	Characterization problems
The fading envelope characteristics	<ul style="list-style-type: none"> • Velocity of the mobile • Frequency of transmission • Fade depth p • Average duration of fade at p • Positive crossing rate of envelope at p 	<ul style="list-style-type: none"> • Given by the experiment's setting • Given by the experiment's setting • NO GUIDELINES FOR SELECTION • Theoretical expressions • Theoretical expressions 	<p>There are no guidelines for the selection of the fade depth p.</p> <p>There is no relation between the method and the average SNR at the receiver, frame size or transmission rate.</p>
Experimental data	<ul style="list-style-type: none"> • Histograms of error free runs • Histograms of error runs 	<ul style="list-style-type: none"> • Experimental traces • Experimental traces 	Experimental data must be available beforehand.

3.3.2. Hidden Markov model construction

For the actual construction of a HMM it is necessary to have as a “hidden” variable, one that follows the Markovian property. Traditionally the SNR has been used as the hidden variable and its Markovian properties have been extensively studied [46][54]. However, SNR values are not available in the experimental traces, therefore it was necessary to use the received signal strength indicator (RSSI) to construct the HMMs. Later in the chapter we will discuss the Markovian properties of the RSSI.

3.3.2.1. HMM construction based on experimental data

To model the frame losses with a HMM it is first necessary to characterize the underlying Markov chain that models the evolution of a Markovian variable. The collected traces do not include the SNR that existed during the reception of each frame but the average received signal strength (RSSI).

The process consists of analyzing the traces and computing a first approximation for the variables π , \mathbf{P} , \mathbf{B} of the HMM. These variables will then represent a “first” model θ that is needed for the Baum Welch algorithm described in Appendix D. With these elements in place it is then possible to generate an output that represents the frame losses. The step by step process is described next.

1. Select the Markovian variable for the underlying hidden process. In our case this is the RSSI available in the traces.
2. Select a partitioning method for the RSSI
 - a. Select the number of states

- b. Select the partitioning scheme
3. Partition the RSSI
4. Obtain first approximation for the values of π , \mathbf{P} and \mathbf{B}

Number of states - This number represents the number of intervals into which the RSSI will be partitioned. During the selection of the number of states, an important fact about the RSSI from the collected traces was first noticed, this is its coarseness.

The RSSI values vary widely from one sample to the next one. It was determined that the information from the traces was only useful to construct Markov models with a low number of states. The reason for this is that as the number of states increases a higher granularity in the recorded data is needed. For example, increasing the number of states to three resulted in observing transitions between non adjacent states. When computing the probability of transitioning between states, these non adjacent state transitions cannot be taken into consideration since the model assumes that transitions occur only within neighboring states. Therefore, after analyzing the RSSI data it was determined that only models with two states can be actually constructed with the data available.

The model under construction does not incorporate transitions to non-neighboring states. The variations in the RSSI could indicate that a model with more than two states and with transitions between non adjacent states is needed. However as it will be shown in later chapters, it was not necessary to use such a model since the results from the two-state one accurately modeled the frame loss process.

Partitioning scheme - As stated in Appendix B, the effects of different partitioning schemes are not critical to the output. Therefore a simple partitioning scheme was selected in which the observed range of the RSSI was divided into two intervals of the same width. The limits of the intervals were computed using the following expressions.

$$\text{For state } s_0: \left[\frac{\max RSSI}{2}, \max RSSI \right]$$

$$\left[\min RSSI, \frac{\max RSSI}{2} \right]$$

For state s_1 :

An outline of the process followed is illustrated in

Figure 17. By analyzing the RSSI in the traces it is possible to count the transitions between the two states and generate the transition probability matrix \mathbf{P} and then the vector π . By looking at the column that indicates if a frame was received in error or error free and the corresponding RSSI state in the Markov chain, it is possible to construct the \mathbf{B} matrices. With these three elements a first model approximation can be obtained and then the Baum Welch algorithm can be used to obtain an “optimal” HMM that represents the observed frame loss process.

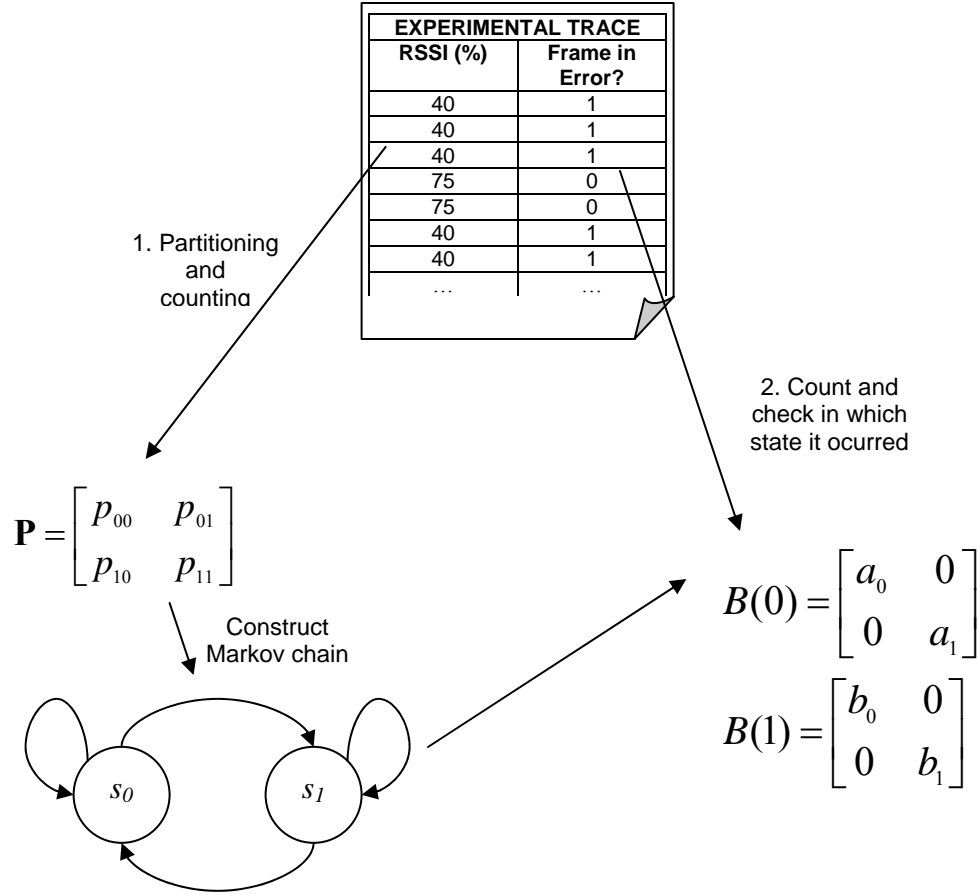


Figure 17 HMM construction of a first model for frame losses based on experimental traces.

In particular, in Figure 17 matrix \mathbf{P} is obtained by partitioning the RSSI and then counting the transitions between states. Each element of \mathbf{P} corresponds to the transition probabilities of the two-state “hidden” Markov process shown on the bottom of the figure.

By looking at the frame error sequence it is possible count how many times a frame arrives in error or is error free during interval of the partitioned RSSI. With this information it is possible to construct the \mathbf{B} matrices. For example, matrix $\mathbf{B}(0)$ contains the probability of observing a “0” in the sequence in each of the two states of the hidden process. In $\mathbf{B}(0)$ the element labeled a_0 represents the probability a “0” occurred in state s_0 ; the element labeled a_1 represents the probability a “0” occurred in state s_1 . In a similar way, matrix $\mathbf{B}(1)$ contains the observation probabilities for observing a “1” in the sequence of frames.

3.3.3. Summary of models constructed for characterizing frame losses

Two types of models have been proposed for characterizing frame losses, a two-state Markov and a hidden Markov model. For the first model two characterization techniques were detailed. Only one characterization technique was detailed for hidden Markov models. The next table summarizes the proposed models and their characterization.

Table 11 Summary of models for characterizing frame losses

Model	Characterization method	Advantages	Limitations
Two-state Markov model	Based on the fading envelope	<ul style="list-style-type: none">Provides a starting point for characterizing a model when no experimental data is available. (However this starting point is not accurate).	<ul style="list-style-type: none">There are no guidelines on how to characterize it given average SNR, frame sizes or transmission rates.State durations are assumed to be geometric.
	Based on experimental data	<ul style="list-style-type: none">Only the state durations are needed for characterization.Experimental data contains information about average SNR, frame size and transmission rate.	<ul style="list-style-type: none">Experimental data must be available.State durations are assumed to be geometric.
Hidden Markov Model	Based on experimental data	<ul style="list-style-type: none">Once a first model is obtained it can be trained using Baum-Welch algorithm and count with a better approximation to experimental data.	<ul style="list-style-type: none">Characterization and implementation include more parameters than a two state model.Experimental data must be available.RSSI granularity limits the number of states for the hidden variable.

3.4. CONSTRUCTION OF MARKOVIAN MODELS FOR TRANSMISSION RATE VARIATIONS

The 802.11 standards allow the variation of the transmission rate of frames based on the observed SNR. These variations could be modeled by constructing a Markov chain or with a HMM that uses the RSSI as the underlying variable for the hidden process. These models would yield as output a variable that

represents the transmission rate and its variations. The selection of the type of model to be used in later chapters will be done after analyzing how well each one fits the experimental data.

3.4.1. FSM construction

The first alternative that was tested for modeling the rate variations was to directly characterize these variations with a Markov chain. In order to construct the model based on experimental data, the rate variations information was extracted from the traces. In particular, the construction of the model consists in computing the transition probability matrix \mathbf{P} for the experimentally observed rates. The values of \mathbf{P} are computed by counting the transitions between the transmission rates.

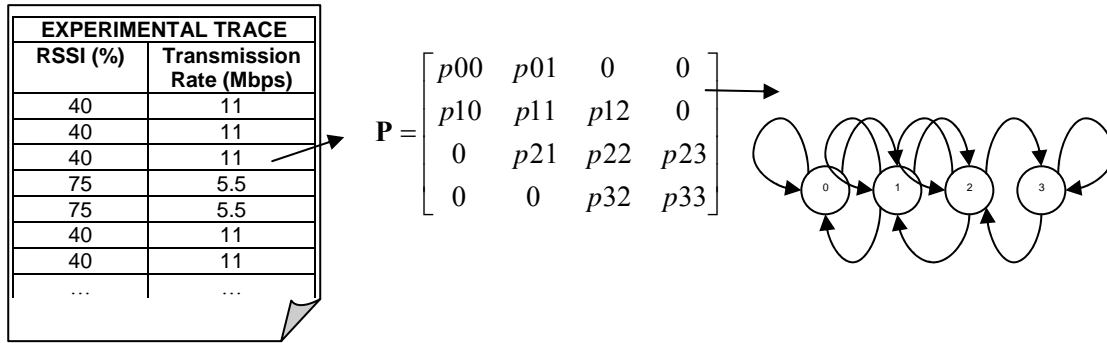


Figure 18 Characterization of rate variations with a Markov model based on experimental data.

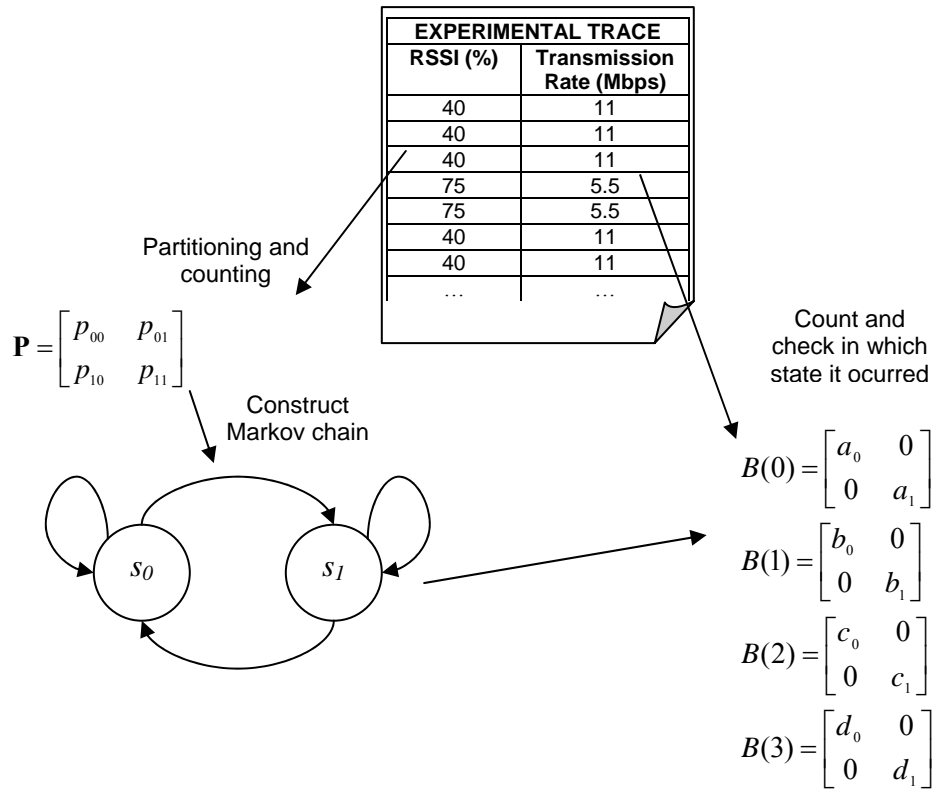
Figure 18 illustrates the process. In this case each of the four states, labeled 0 through 3, represent an individual transmission rate. The process was constructed as a birth-death process, which only includes transitions between neighboring states. To deal with non-adjacent transitions, while maintaining simplicity, if the experimental trace indicated that a transition between non neighboring states occurred, it was assumed that a transition to an intermediate state occurred first.

Times in intermediate states generated by non-adjacent transitions correspond to the duration of only one frame. This duration of a frame is extremely small compared to how much time the system stays at any transmission rate. For example a maximum sized frame (1500 bytes) transmitted at the lowest transmission rate (1Mbps) takes only 12 ms to be transmitted. On the other hand an 802.11 system selects any transmission rate for more than a few seconds, sometimes even dozens of seconds. Therefore

counting transitions to intermediate states and including them in \mathbf{P} will not affect the characterization process.

3.4.2. HMM construction

The generation of transmission rates as the output of the HMM can be done by taking the RSSI, partitioning it to obtain a first model, and then using the Baum Welch algorithm to obtain an optimal model. This procedure is basically the same to that one illustrated in Figure 17. However, instead of considering the frame loss information from the traces, the \mathbf{B} matrices are constructed by analyzing the transmission rates. Figure 19 illustrates the procedure.



$B(0)$ represent the 11 Mbps periods, $B(1)$ the 5.5 Mbps, $B(2)$ the 2 Mbps and $B(3)$ the 1 Mbps.

Figure 19 HMM construction of a first model for rate variations based on experimental traces.

In Figure 19, matrix \mathbf{P} is obtained by partitioning the RSSI and then counting the transitions between states. Each element of \mathbf{P} corresponds to the transition probabilities of the two-state “hidden” Markov process shown on the bottom of the figure.

By looking at the transmission it is possible to count how many times a particular rate is selected during each interval of the partitioned RSSI. With this information it is possible to construct the \mathbf{B} matrices. For example, matrix $\mathbf{B(0)}$ contains the probability of observing a frame transmitted at 1 Mbps in the sequence in each of the two states of the hidden process. In $\mathbf{B(0)}$ the element labeled a_0 represents the probability a transmission rate of 1 Mbps occurred in state s_0 ; the element labeled a_1 represents the number of a transmission rate of 1Mbps occurred in state s_1 . In a similar way matrices $\mathbf{B(1)}$, $\mathbf{B(2)}$ and $\mathbf{B(3)}$ contains the observation probabilities for transmissions at 2 Mbps, 5.5 Mbps and 11 Mbps respectively.

3.4.3. Summary of models constructed for characterizing rate variations

In this section, two types of models have been proposed for characterizing rate variations, a multi-state Markov and a hidden Markov model. In the Markov model each state represents a transmission rate. In the HMM the RSSI is used to follow the transmission rate. The next table summarizes the proposed models and their characterization.

Table 12 Summary of models for characterizing rate variations

Model	Characterization method	Advantages	Limitations
Multi-state Markov Model	Based on experimental data	<ul style="list-style-type: none"> Only the state durations are needed for characterization. Experimental data contains information about average SNR, frame size and transmission rate. 	<ul style="list-style-type: none"> Experimental data must be available. State durations are assumed to be geometric.
Hidden Markov Model	Based on experimental data	<ul style="list-style-type: none"> Once a first model is obtained it can be trained using Baum-Welch algorithm and count with a better approximation to experimental data. 	<ul style="list-style-type: none"> Experimental data must be available. RSSI granularity limits the number of states for the hidden variable.

3.5. COMPOSITE MODEL CONSTRUCTION

In 802.11 systems both frame losses and rate variations are present at the same time. An accurate way of modeling the effects of the channel on communications should include both effects. It is possible to construct a composite model that includes these effects by combining the models proposed before.

The results obtained from the experimental traces showed that one of the processes, the transmission rate variations, possesses states durations that are much longer than those from the frame loss process. For instance, the duration of a transmission rate state is usually in the order of dozens of seconds while the duration of frame loss condition is in the order of seconds. By taking into account this characteristic a composite model is proposed next.

The construction of a model for both processes can be made by combining the frame traces generation that uses a HMM based on the variation of the RSSI, with a Markov chain that follows the variations of the transmission rates. Figure 20 illustrates the generic structure of such a model.

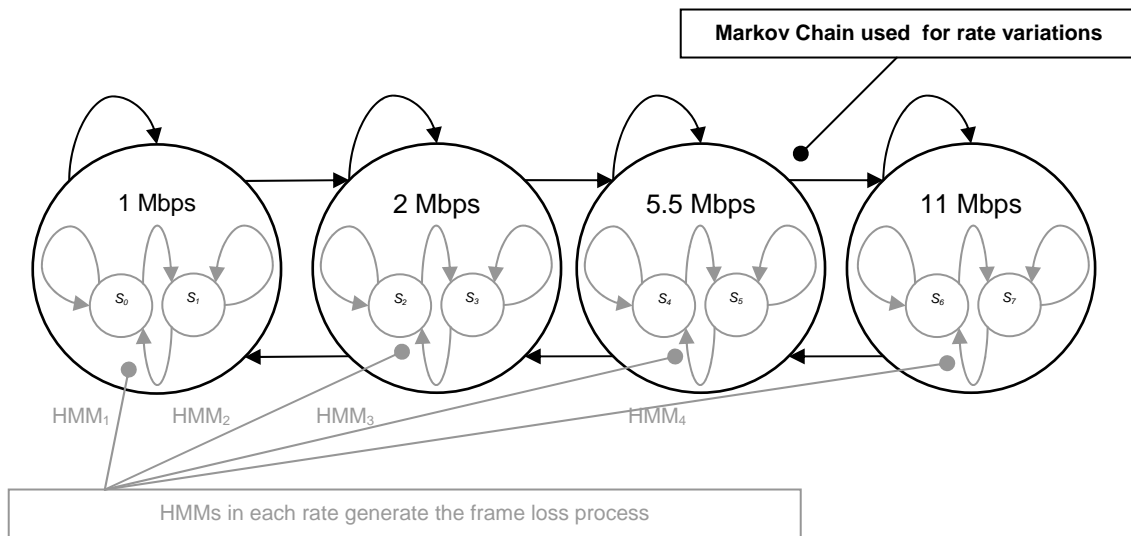


Figure 20 Composite model construction.

The construction in Figure 20 shows the composite model for an 802.11b system. The model contains a Markov chain the follows the transmission rate variations. This chain is the same used to

model the rate variations in the previous section. Therefore the duration of each state in the chain represents the length of time the system transmit data at each rate.

During each state, the frame loss process is generated by activating a HMM model. Each rate has a different HMM that should be characterized with the procedure detailed in Section 3.3.2.1. When the transmission rate changes, a new HMM is selected and the corresponding frame loss process gets activated. A similar solution can be used for 802.11a systems by including the corresponding transmission rates.

3.6. VALIDATION OF FRAME LOSS MODELS

The output of the models that can be constructed using the methods described in the previous section, is a discrete variable that represents either the frame loss process or the transmission rate variations. This section details the methods for analyzing the validity of these output variables. The validity will be studied via quantitative methods that indicate how close the model is to the experimental results. One of these methods, the Kolmogorov-Smirnov test, described in the next subsection will be used in both the analysis of the output obtained with FSMC models and hidden Markov models. Another method, the contingency table analysis, will be used in the study of the Markovian property for hidden Markov models.

3.6.1. Validation of the frame loss FSMC model output

When a FSMC model is used to represent the frame loss process over the channel, the output is a binary variable that indicates if a received frame is in error or not. In order to compare how similar the distribution of this output variable is to the experimental traces, a Kolmogorov-Smirnov test can be used. This test is described next.

3.6.1.1. The Kolmogorov-Smirnov test for two independent samples

The distributions of error and error free runs generated by a model can be compared to those observed experimentally by using a Kolmogorov-Smirnov test (also called K-S test) for two independent samples. This test is nonparametric [30], meaning that it makes no assumptions about the underlying distributions of the data to be compared. For the run distributions this test is particularly useful since the experimental data may not fit any known distribution. Additionally, neither the model's run distributions

nor the experimentally observed run distributions appear to be normal, a requisite for some parametric tests such as the t-test.

The two-sample K-S test is based on the comparison of two sample distribution functions. The comparison is based on the statistic D [30] defined as:

$$D = \sup_{\text{all } x} |F_m(x) - G_n(x)|,$$

Where $F_m(x)$ is the sample distribution function of a sample of size m from X and $G_n(x)$ is the sample distribution function of a sample of size n from Y . The hypothesis of equal distributions is rejected, at a significance level α , when:

$$D > \left[-\frac{1}{2} \left(\frac{1}{m} + \frac{1}{n} \right) \log \frac{\alpha}{2} \right]^{\frac{1}{2}}.$$

3.6.1.2. Validation of the FSMC model output against experimental data

In order to validate the output coming from the FSMC model it is first necessary to use the method illustrated in Figure 16 to characterize the model and then use it to generate frame losses over a channel. Such a basic setup is illustrated in Figure 21.

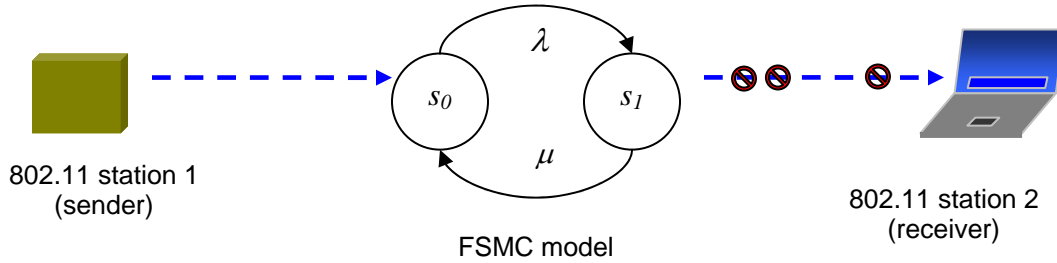


Figure 21 Setup required to validate the FSMC model output.

By selecting the appropriate transmission rates and frame sizes at the sender in Figure 21, a trace that can be directly compared to the corresponding experimental trace is generated. The similarity between the two traces, the model's output and the experimental one, can then be analyzed by using a two sample K-S test. This test can be applied to the error and error free runs separately to compare their distributions to the ones from the experimental traces. As result the two-sample K-S will indicate if these variables have the similar distributions or not.

3.6.2. Validation of the assumption behind the hidden Markov model construction

In order to construct the hidden Markov model using the parameters defined in Section 2.1.2.1 it is first necessary to verify if the ‘hidden’ variable follows the Markovian property or not as explained in Appendix B. The RSSI is the only variable available in the experimental traces that represents the channel’s quality and it is necessary to use it in the construction of the HMM. Therefore it is relevant to first test the Markovian characteristic of the RSSI. For this Markovian test the method of contingency tables can be used.

3.6.2.1. Contingency table construction for validating the Markovian assumption

The Markovian property indicates that in a Markov process, the probability of transition at a time ‘n’ to a new state depends only on the state at time ‘n-1’. Contingency tables can be used to verify if experimental data follows this property. In particular this is useful in the analysis of the RSSI values captured in the experimental traces.

In general, a contingency table summarizes the results of classifying data. For example, when one performs an experiment, each outcome can be classified according to more than one variable and the results can be presented in a table. The table represents an array of frequencies in which each cell stands for the count of the row and column intersection. When only two variables are considered, this table is called a *two-way contingency table* [20] [30].

For the Markovian study of the RSSI it is first necessary to partition the RSSI range in two states; the actual partitioning details are detailed in the Section 3.3.2. As the RSSI value changes one can record in which state the RSSI falls in. With this information, contingency tables can be constructed by taking into consideration the current and previous states of the RSSI sequence.

The simplest contingency table that can be constructed is presented next. This table shows as column and row headings the current and previous states of the RSSI evolution in a collected trace. For this example, the table illustrates the number of times each two state combinations occurs.

Table 13 Two-way contingency table for the RSSI and a previous state sequence length of one.

Current State	Previous State		Totals
	0	1	
0	a	c	a + c
1	b	d	b + d
Totals	a + b	c + d	T=a + b + c + d

The count in each individual cell (labeled a, b, c and d) represents the number of times the observed RSSI went from the state indicated by the intersection of the column labeled “previous state” with the row labeled “current state”. In the particular case of the study of the Markovian property, one is interested in analyzing the independence or dependence between the previous and current state.

To study the independence or dependence between the variables, a chi-square test of independence [20] can be used. For this test the null-hypothesis is that the two variables have no relationship; this would mean that the current state does not depend on the previous state.

To perform the chi-Square test of independence one can compute the “expected cell frequencies” using the expression:

$$E_{ij} = \frac{Total_i \times Total_j}{T},$$

where E_{ij} represents the expected frequency for the cell in the i th row and the j th column. $Total_i$ represents the total number of transitions in the i th row and $Total_j$ is the total number of transitions in the j th column. T is the total number of transitions in the whole table. For the example above, four values of E_{ij} can be computed, one for each cell. With these values it is possible to compute the chi-square value for the contingency table, which is expressed as:

$$\chi^2 = \sum \frac{(\text{expected} - \text{observed})^2}{\text{observed}}$$

$$Degrees_of_freedom = (Rows - 1)(Columns - 1)$$

With the chi-square value it is then possible to compute the corresponding probability value from the chi-square distribution and either reject or accept the null-hypothesis. If this results in a probability value of less than 0.05 (95% confidence level) it means that the null-hypothesis is rejected and that the current state depends on the previous state. Notice that this dependency is expected to be observed in data that follows the Markovian property.

The independence or dependence between the current state and several previous states can be analyzed in a similar way. This is done by constructing tables with previous state sequences of length greater than one. For example, the next table shows the counts for the transitions to the current state occurring from a history of two previous states.

Table 14 Sample Two-way Contingency table for the RSSI and a previous state sequence length of two.

Current State	Previous State Sequences				Totals
	00	01	10	11	
0	m	o	q	s	m + o + q + s
1	n	p	r	t	n + p + r + t
Totals	m + n	o + p	q + r	s + t	m + n + ... + t

The chi-square test of independence can be applied again to the last table. In this case, with three degrees of freedom, if this results in a probability of less than 0.05 the null hypothesis is rejected. This would mean that the current state does have a strong dependence on the two previous states. If the value of computed chi-square is higher than in previous state sequence length of 2, this could mean that this dependence starts to decrease.

During the actual data analysis, the evolution of the RSSI was used to construct two-way contingency tables. The results for previous state sequences lengths of two, three and four indicated that the RSSI process was not Markovian since a dependency with the previous states was observed. However, as the length of previous state sequence increased, the dependency between the previous states and the current state decreased significantly. Therefore the representation of the RSSI sequence with a two-state first order Markov chain is only an approximation of the experimental sequence.

Since the RSSI is the only value available in the experimental traces that links the received frame with the SNR present at the time of reception, it had to be included in the creation of HMMs. As it will be illustrated in chapter 6, the results obtained from the models for frame losses do not differ significantly from those observed in the experiments. The similarity between the simulation results and the experimental observations validate the coarse approximation of the RSSI with a first order Markov chain for the particular conditions of the experiments.

3.6.3. Validation of the hidden Markov model frame loss output

The validation of the HMM output can be performed in the same way as in the FSMC case. When used to model frame losses, the HMM output is also a binary variable. The error and error free runs from the output can be analyzed with the same two-sample K-S test used in for the FSMC output analysis. Unfortunately, the tool used to collect the experimental traces does not allow the acquisition of the actual average SNR value during the reception of individual frames. As explained before the RSSI value is

recorded by the tool and therefore it is first necessary to verify if this variable follows or not the Markovian property.

3.6.4. Validation of frame loss processes by analyzing the effects on upper layers

Besides analyzing the output from the models with two sample K-S tests it is possible to analyze the effects of the models on upper layers. These effects can be compared with those observed from the experimental frame losses. The loss of frames at the MAC layer level could have effects on upper layers. In particular, transport layer protocols like TCP can get affected by frame losses. For example, a TCP sender can misinterpret as congestion, the timeouts that occur due to information segments lost due to the channel conditions. This misinterpretation leads to periods of inactivity during which no information is sent over the channel even if the channel is in a good state.

In order to further compare the modeled frame loss processes with the experimental traces, it is possible to analyze the effects of the frame losses on TCP/IP data transfers. In particular, the effects on file transfers can be studied by using the model in a setup that incorporates the file transfer protocol (FTP). The setup should be able to use both the constructed models and the experimental data as frame loss processes. By comparing the effects on transfer time of files it is possible to further understand the impact on upper layers coming from the output of the model and the actual experimental traces. The actual setup used for this comparison will be described in Section 3.10.

3.7. VALIDATION OF THE MODEL FOR THE TRANSMISSION RATE VARIATIONS

A transmission rate variations model has as output a discrete variable with levels that represent each transmission rate available for a particular technology. The duration of each transmission rate and its distribution can be studied with the same methods described in the last section. In particular, for understanding the similarities or differences between the distributions it is possible to use the two-sample K-S test. In this case, the run for each transmission rate can be compared to those observed in the experimental traces. Additionally it is possible to compare the mean transmission rate during the transfer of frames; this comparison will be performed using a t-test.

3.8. VALIDATION OF THE COMPOSITE MODEL

The outputs of the composite model are two variables that represent the transmission rate and the frame loss process. This research will validate these variables separately using the methods from sections 3.6 and 3.7. In the final chapter we will discuss the limitations and challenges of validating the composite model as a whole in the case one desires to study the effects of the model on upper layers.

3.9. VALIDATION SUMMARY

The next table summarizes the general validation methods that will be used.

Table 15 Summary of validation techniques

Validation technique	Process being validated	Model under testing
<ul style="list-style-type: none">• Two sample K-S test between model's output and experimental data for distributions• t-Test for mean duration of runs	Frame losses	FSMC characterized with data form the fading envelope.
	Frame losses	FSMC characterized with experimental data.
	Frame losses	HMM characterized with experimental data.
	Transmission rate variations	HMM or FSM
	Frame losses and transmission rate variations	Composite Model
Two way contingency tables	Markovian property of RSSI	HMM hidden process validation

3.10. SIMULATION PLATFORM AND GENERAL IMPLEMENTATION FEATURES

The validation processes detailed in this chapter were implemented using computer simulations. Two main tools were used for such implementations, Matlab (v6.5) and Opnet's Modeler (v10.5p11). Matlab was also used for all the statistical analysis along with SPSS (v11). All the software tools were run under Windows and/or Solaris operating systems.

In particular, for the generation of experimental traces such as those that can be obtained with the setups from Figure 21, Matlab or Modeler were used to analyze the experimental traces and then create the FSMC or HMM outputs. A major advantage of implementing these models inside Modeler is that this

tool allows easy testing of the model's in packet switched wireless networks configurations similar to that illustrated in Figure 23.

All the simulations have a common implementation strategy for the generation of transitions between states. To illustrate this concept, Figure 22 shows state “i” of a Markov chain with its associated transition probabilities $p_{i, i-1}$, $p_{i, i+1}$ and $p_{i, i}$. Notice that each of these transitions corresponds to row “i” of the transition probability matrix \mathbf{P} . Therefore, if the current state in a simulation is “i”, the next state is decided by generating a uniformly distributed random number “r” between zero and one and searching for the first index that would make the following condition true:

$$r < \text{cumulative sum of row “i”}$$

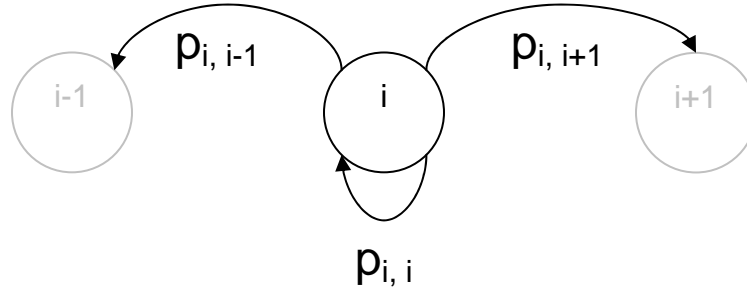


Figure 22 A state of a Markov chain with its associated transition probabilities.

The cumulative sum of row “i” is a vector that is formed by adding up the elements of a row up to each column index. In this particular case, if row “i” of matrix \mathbf{P} is a vector of the form:

$$[0 \quad 0 \quad 0 \dots p_{i, i-1} \quad p_{i, i} \quad p_{i, i+1} \quad \dots \quad 0 \quad 0]$$

Then the cumulative sum vector takes the form:

$$[0 \quad 0 \quad 0 \dots p_{i, i-1} \quad p_{i, i-1}+p_{i, i} \quad p_{i, i-1}+p_{i, i}+p_{i, i+1} \quad \dots \quad p_{i, i-1}+p_{i, i}+p_{i, i+1}]$$

After this the random value “r” is compared to each element of the cumulative sum vector and when the above condition is true, the index of that element is selected as the next state. In Matlab the command `cumsum` and `find` give this index directly by using them as in the following construction:

```
next_state = min(find(rand(1) < cumsum(vector)))
```

Where `rand(1)` generates a uniformly distributed number between zero and one and `vector` is an array that contains the probability vector. This same strategy can be used for the generation of outputs like frame loss processes based on probability vectors.

3.10.1. General simulation setup in Modeler

It is not possible to capture all the characteristics of an experiment in simulation. In general a complex simulation may take into consideration several variables that may result in increased accuracy. However complex aspects such as processing time at stations or propagation characteristics can be very time consuming to simulate. For this reason when the effects of the model on upper layers are studied, it is possible to compare the model's effect with a trace driven simulation.

The trace information would tell a receiver when to accept or discard a frame. An OPNET's Modeler (v10.5) simulation can be implemented to perform such tasks. To understand the model's effects on FTP transfers a computer simulation was constructed using the basic setup illustrated in Figure 23. This setup consists of a sending 802.11b station and an 802.11b receiver. Both stations had a complete 802.11b and TCP/IP stack.

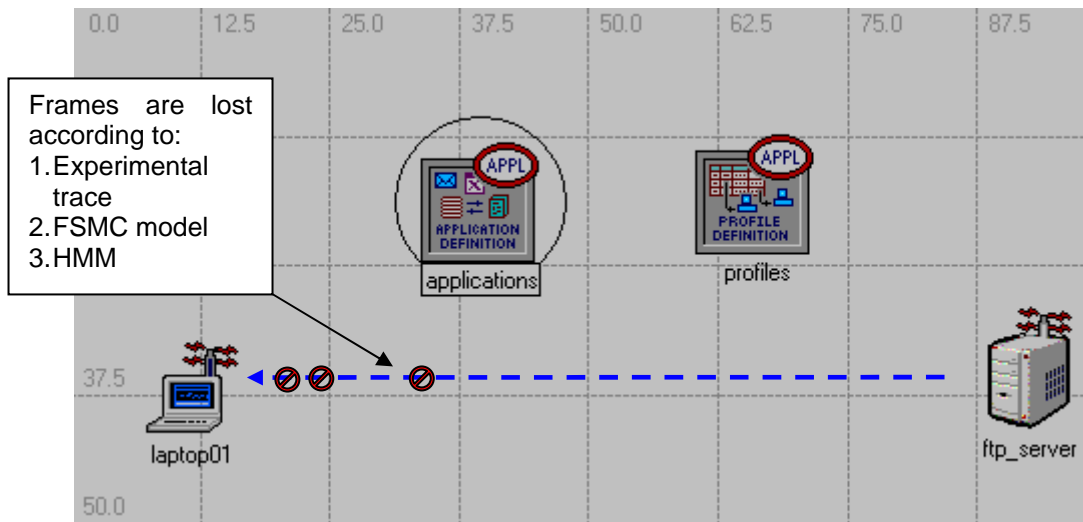


Figure 23 Simulation setup for comparing the frame loss traces effects on an FTP file transfer.

In the simulation a wireless client (laptop01) requests via a file transfer protocol (FTP), five 100 Mbytes files from the FTP server. The server responds by allowing a TCP connection for downloading the files. The client implemented three different mechanisms for losing arriving frames: based on an experimental trace, a FSMC model and a HMM. Only one of these three mechanisms was activated at any time.

Since the simulation tool allows the configuration of the frame size and transmission speed used during the file transfer, it is possible to match these conditions to those under which the experimental traces were collected. With this setup it is therefore possible to compare the time transfer of the files using the three frame loss mechanisms.

4. EXPERIMENTAL OBSERVATIONS AND ANALYSIS

4.1. EXPERIMENTAL RESULTS OF FRAME LOSS PROCESS

The traces collected during the experiments generated a considerable amount of data (approximately 4 Gigabytes). For this reason it is essential to summarize the effect of each factor on the error distributions. The effects can be illustrated by varying one of the three factors (i.e. average SNR at the receiver, transmission rate and frame size) while keeping the other two constant. Variations in the average SNR at the receiver are accomplished by moving the receiver from location to location. Transmission rates and frame size variations were manually configured parameters at the sending station. Figure 24 illustrates how the results will be presented in the following sections for both 802.11b and 802.11a technologies. The results for office and residential environments will be presented separately.

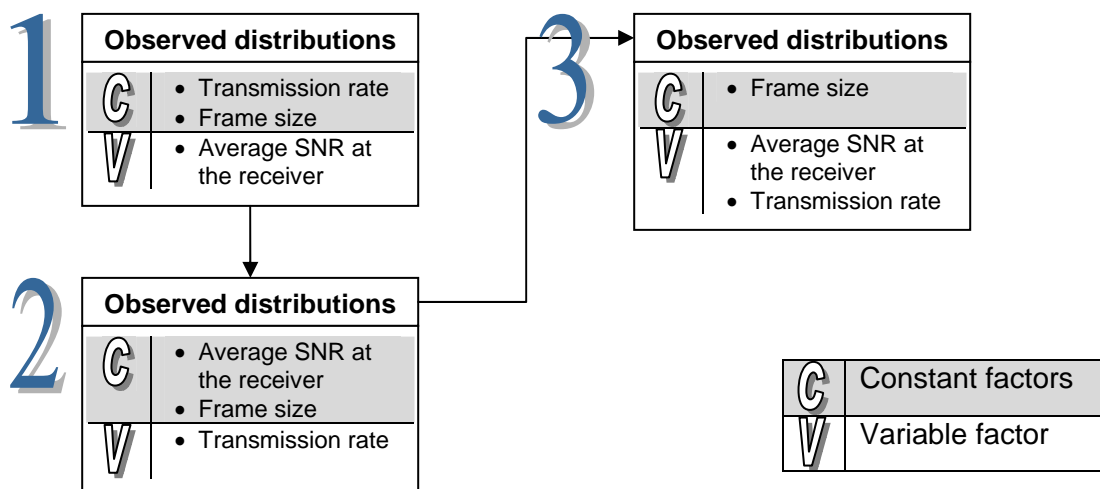


Figure 24 Presentation order of the experimentally observed distributions

4.1.1. Frame loss distributions observed with 802.11b

In this section we consider 802.11b systems at 2.4 GHz with four transmission rates. By keeping both the transmission rate and the frame size fixed, it is possible to observe the effect on the distributions of moving the receiver farther away from the transmitter. In general the next results will illustrate how under the highest average SNR conditions tested, the experimentally observed distributions presented an exponential shape. Error free runs usually allowed thousands or tens of thousands of frames to be transmitted before being interrupted by very short (1 or 2 frames) error runs that occur very sporadically. When the average SNR at the receiver was the lowest the opposite effect was observed.

4.1.1.1. Office environment

Figure 25 illustrates the effect of different average SNR at the receiver on the error distribution in the office environment. The left part of the figure shows the distribution for the error free runs, while in the right the ones for the error runs are plotted. For this experiment, the average SNR at the receiver varied from 36dB to 23dB, for locations **a** and **d** in Figure 9 respectively. The transfer consisted of approximately 100,000 frames of 1,500 bytes each at 11Mbps. On the left part of the figure a grayscale legend is included to help with the understanding of this and subsequent plots. *The legend changes brightness as the average SNR (or other factor) varies. Darker shades indicate higher the levels for the factor under study.*

The effect of the different locations on the distribution can be seen on the tail length of the error and error free runs. When going from high average SNR to low average SNR, the long tails observed in the error free distribution get shorter. This is also reflected in the percentage of frames received in error, 0.21% and 84.19% for locations **a** and **d** respectively. These general visual observations about the loss distributions suggest that a simple exponential model could be used to generate error and error free periods. However, a qualitative test is necessary before making this claim. The results from such a test are presented in the next chapter.

The effects of varying the transmission rate between the stations while keeping constant the average SNR at the receiver and the frame size can be observed in Figure 26. In this case all results shown correspond to location **d** in which an average SNR of 23dB was measured. These results correspond to transfers of 1,500 byte frames. The percentages of frames received in error at this location were 0.008, 0.044, 12.17 and 84.19% at 1, 2, 5 and 11Mbps respectively.

In Figure 26 it can be seen that for 1 and 2 Mbps transfers, which had a percentage of frames received in error less than 0.05%, the distribution of the error and error free runs does not have an exponential shape. Under these conditions the frames received in error are seen at extremely sporadic

intervals of time. For example the number of frames received in error was three (out of 38,000) at 1 Mbps and seven (out of 15,000) at 2 Mbps.

At higher transmission rates the exponential shape for both distributions was again observed. As the transmission rate increases the tail of the error free run distributions decreases, while the tail of the error run distributions increases considerably.

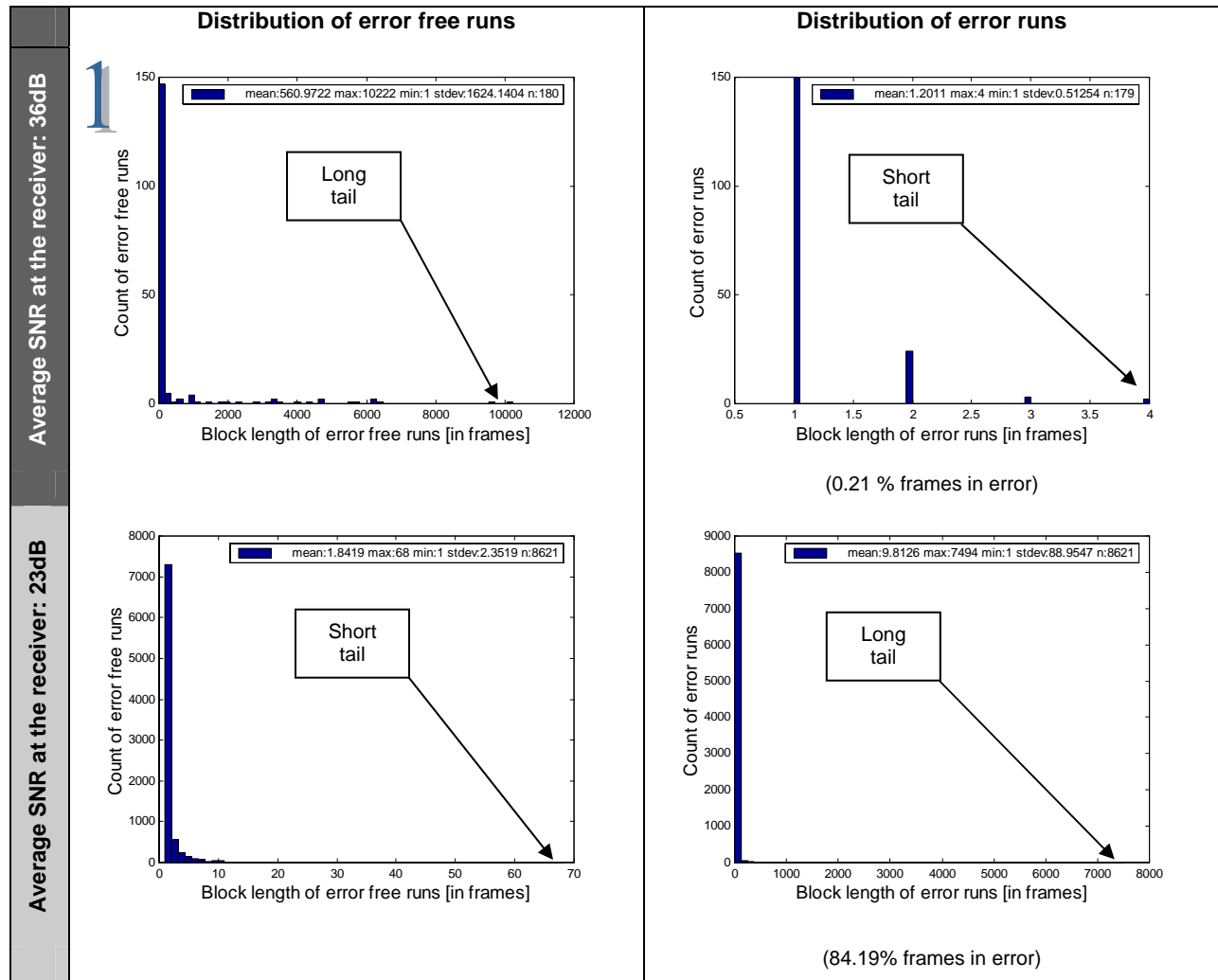


Figure 25 A sample of experimentally observed frame loss distributions at two different office locations. (802.11b, average SNR at location **a**=36dB, average SNR at location **d**=23 dB, frame size=1500 bytes, tx rate = 11 Mbps)

The effects of varying the frame size while keeping constant the measurement location and the transmission rate are shown in Figure 27. These figures were generated from traces collected at 11 Mbps at an average SNR at the receiver of 23 dB (location **d**). The frame size was varied from 1500 to 100 bytes per frame.

The variations in frame size did not change the shape of the distributions. For all the cases in Figure 27 the shape of the histograms is similar to that of an exponential distribution. The percentage of frames received in error varied from 25.48% for 100 byte frames to 84.19% for 1,500 byte frames. This variation is what was expected, since larger frames have a higher probability of getting corrupted in transit in comparison to small frames.

4.1.1.2. Residential Environment

The results obtained after the analysis of the experimental traces at a residential site showed the same tendencies as those presented in the previous section. In this section only the observed distributions at different locations with the transmission speed and frame size fixed will be presented. Further information can be found in the Appendix. Figure 28 shows the distributions for a transfer of 100,000 frames at 5.5 Mbps with 1000 byte frames. The data was collected in a residential environment at locations **p** and **q** from Figure 9.

Figure 28 shows how as in the same manner as in the office environment, the error free run tails get shorter as the average SNR at the receiver decreases. On the other hand, and as in the office environment, the tails of the error runs increase with the average SNR decreases. Both the histograms for the error and error free runs have exponential shapes again visually suggesting that an exponential distribution could be used to model them.

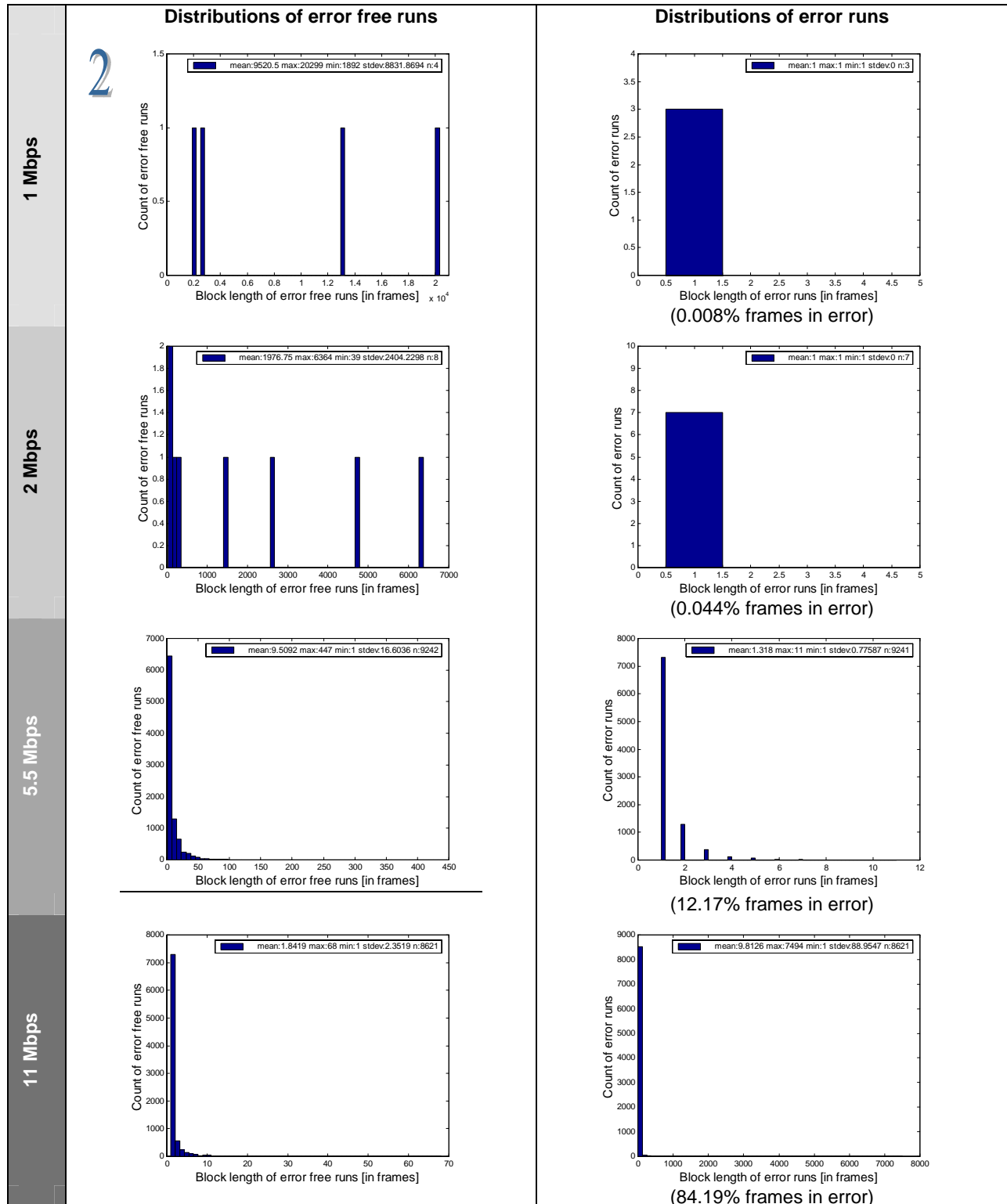


Figure 26 A sample of experimental frame loss distributions at 1, 2, 5.5 and 11 Mbps at office location **d**.
(802.11b, average SNR at location **d**=23dB, frame size=1500 bytes)

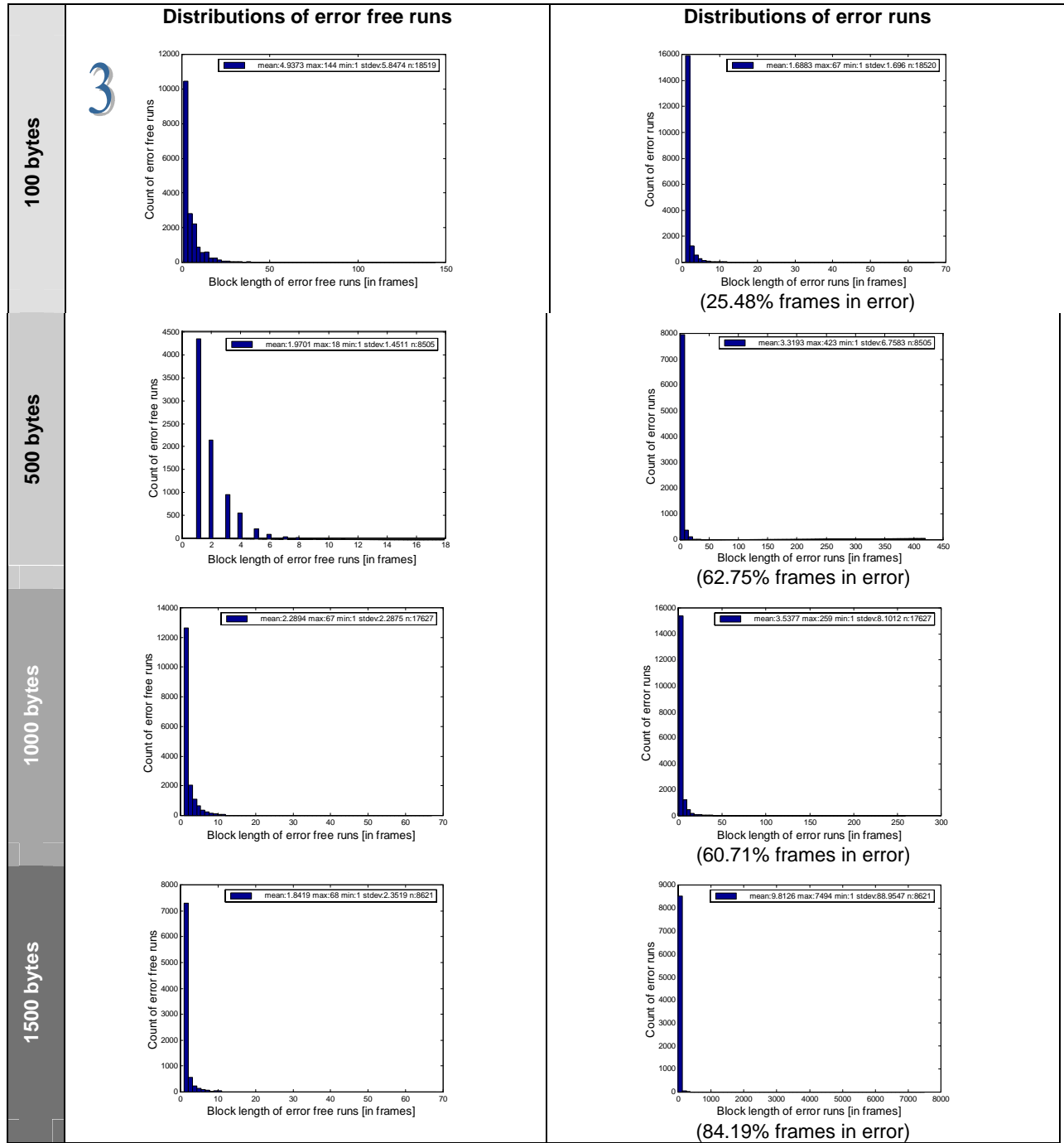


Figure 27 Sample experimentally observed frame loss distributions for various frame sizes at office location **d**.
(802.11b, average SNR at location **d**=23dB, frame size=1500 bytes, tx rate=11 Mbps)

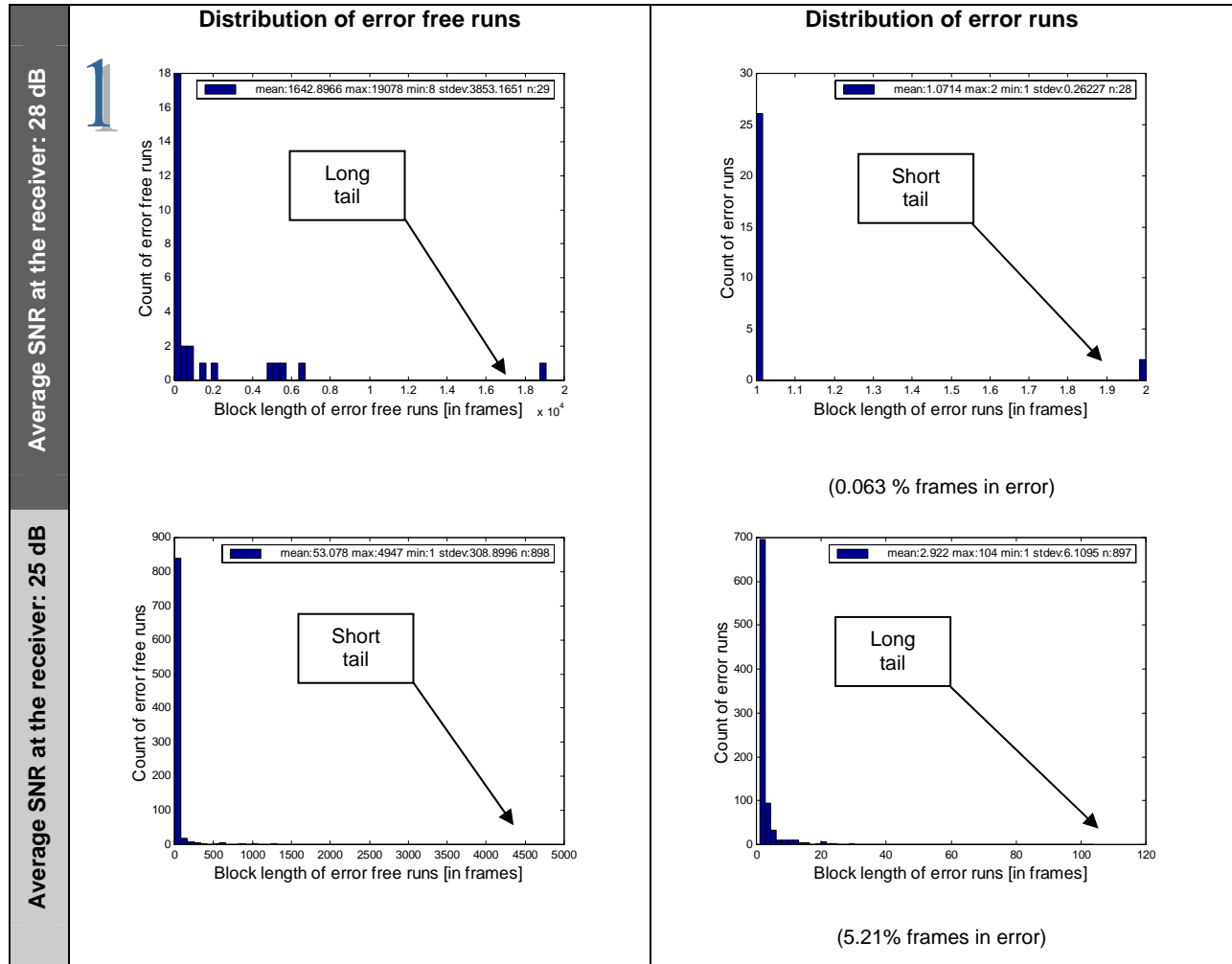


Figure 28 A sample of experimentally observed frame loss distributions at two different residential locations. (802.11 b, average SNR at location **p**=28 dB, average SNR at location **q**=25 dB, frame size=1500 bytes, tx rate=11 Mbps)

4.1.2. Frame loss distributions observed with 802.11a

4.1.2.1. Office environment

Locations **a** through **c** were also used to collect the experimental traces using 802.11a technology. In this case, from all the available transmission rates only 6, 12 and 24Mbps were tested. For the remaining rates the collection of data was not possible since the software tool available was not able to generate useful data for the analysis.

Figure 29 shows the distributions observed at different measurement locations. The average SNR at the receiver was 17, 14 and 9dB for locations **a**, **b** and **c** respectively. The frame size and transmission rate was kept constant during the transfer of approximately 100,000 frames at 24Mbps. The frame size used for obtaining the distributions in the figure was 1500 bytes.

Moving the capturing station from location **a** towards location **c**, increased the percentage of frames received in error from 3.1 to 61.7%. At location **d** it was not possible to obtain usable data since the frame loss percentage was close to 100% and the traces contained incoherent data. The figure also illustrates how the same expected effect on the distributions is observed. In particular at high average SNR the error free distributions have longer tails than at low average SNR conditions.

The effect of varying the transmission rate at location **c** is illustrated in Figure 30. At this place, the average SNR was 9 dB. The effects on the distribution of transferring 1500 byte frames at 6 and 12 Mbps is similar to what was observed in the 802.11b case. Both distributions have an exponential shape. The percentages of frames in error were 21.41 and 94.73% at 6 and 12 Mbps respectively. Notice how this last percentage is higher to that one observed in an 802.11b system at higher average SNR (see Figure 26 at 11 Mbps and 23 dB). These differences in the percentage at similar speeds are due to the disparities in propagation characteristics and modulation schemes between 802.11b and 802.11a technologies.

Figure 31 shows the effects of varying the frame size from 500 to 1000 bytes. At location **c** both runs had exponential shapes for their distributions. Taking measurement with 100 byte frames was not always possible with 802.11a technology since the software tool would frequently generate incoherent data. As in the case of 802.11b systems the results for a residential setup resembled those of the office case.

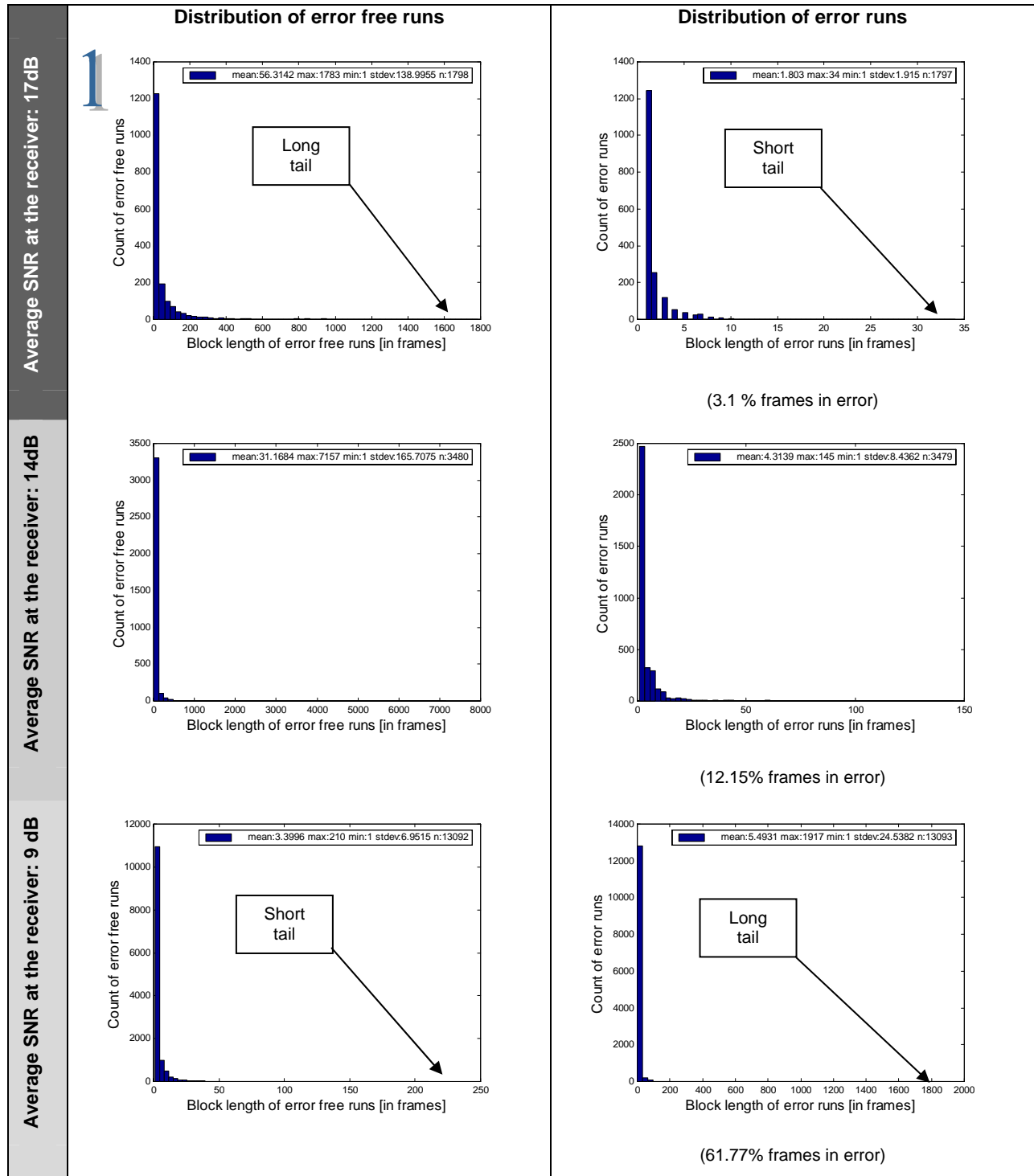


Figure 29 A sample of experimentally observed frame loss distributions at 2 different office locations. (802.11a, average SNRs 17, 14 and 9 dB for locations **a**, **b** and **c** respectively, frame size=1500 bytes, tx rate=24 Mbps)

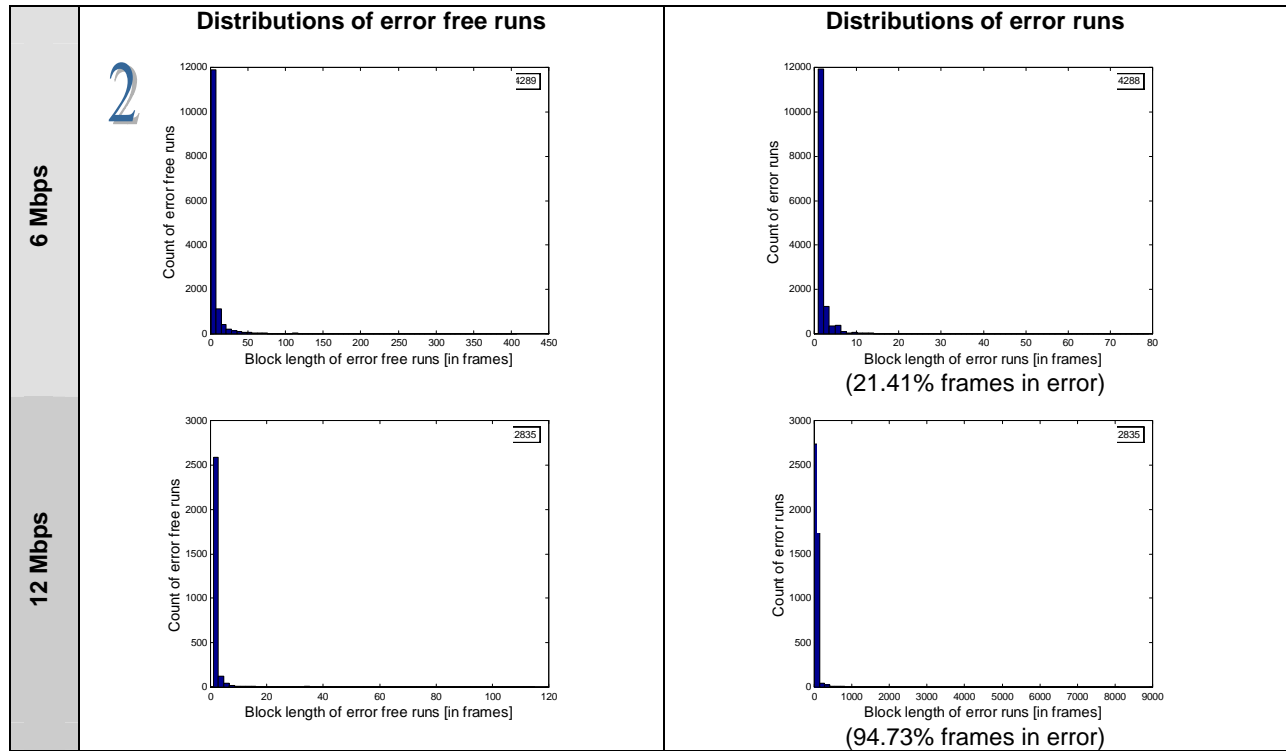


Figure 30 A sample of experimentally observed frame loss distributions at 6 and 12 Mbps at location **c**.
(802.11a, average SNR at location **c**=9 dB, frame size=1500 bytes)

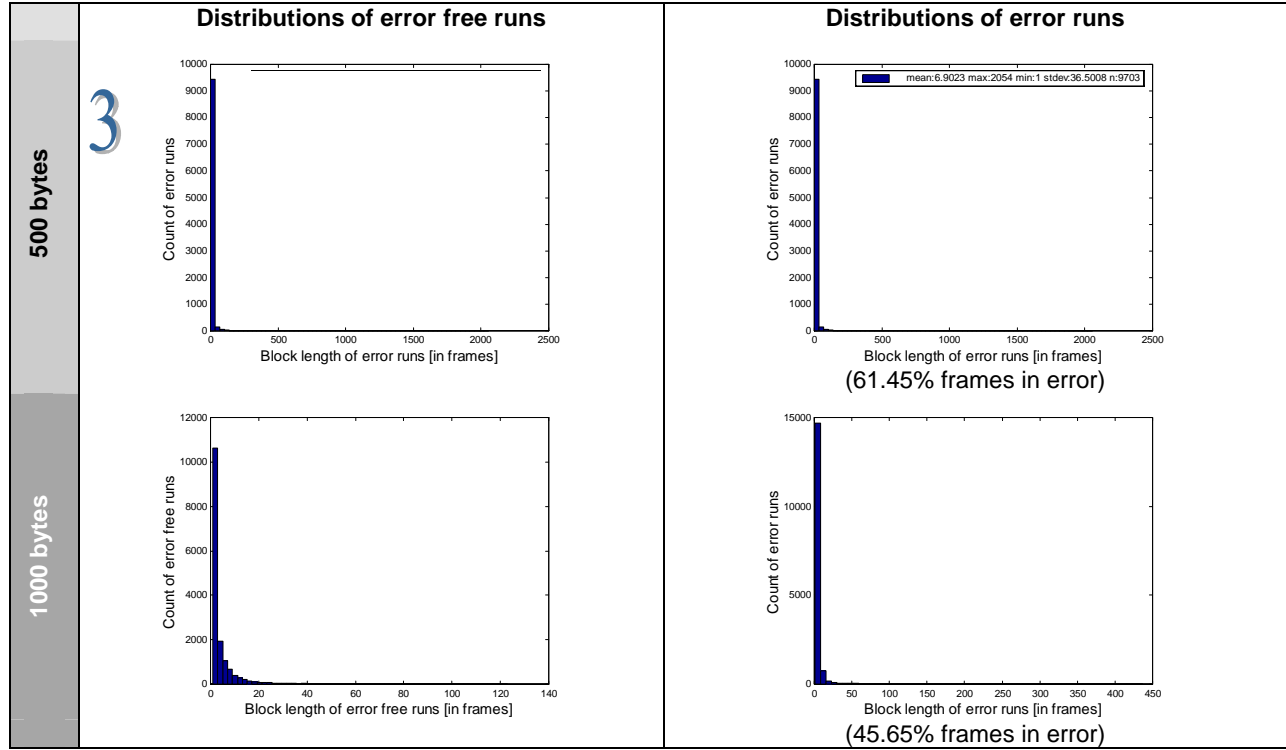


Figure 31 Sample experimentally observed frame loss distributions for various frame sizes at office location c.
(802.11a, average SNR at location c=9 dB, frame size=1500 bytes, tx rate=24 Mbps)

4.1.3. Summary of results for 802.11b and 802.11a systems

The experimental results for 802.11b and 802.11a systems showed the same tendencies for both sites, office and residential. In particular at both sites, as the average SNR at the receiver decreased the error free run distribution showed shorter tails while the error run distribution showed longer tails. This is because at lower average SNR the percentage of frames in error increases, therefore long periods of consecutive frames in error are likely to occur.

The shape of the distribution of the error free and error runs also changed with the transmission speed. This was observed for both 802.11b and 802.11a systems. At lower transmission rates the percentage of frames in error is lower than at higher rates and the runs do not show exponential shapes at high average SNR conditions.

Another common characteristic between the 802.11b and 802.11a results is the variation of the percentage of frames in error with the frame size. As the frame size increases the percentages of frames

received in error also increases. This is what was expected, since large frames have higher probability of getting corrupted in comparison to small frames.

Table 16 Summary of results for 802.11b and 802.11a systems

Characteristic	Observation	Comments
Tail of error free runs	Shortens as the average SNR at the receiver decreases	The maximum samples that contribute to long tails do not have significant values of cumulative frequencies. In particular the samples that create long tails contribute 1% or less in cumulative frequency.
Tail of error runs	Grows as the average SNR at the receiver decreases	The maximum samples that contribute to long tails do not have significant values of cumulative frequencies. In particular the samples that create long tails contribute 1% or less in cumulative frequency.
Distribution of error free and error runs at low transmission rates	Do not have an exponential shape under high average SNR and low transmission rates. As the transmission rate increases the exponential shape appears.	When operating at low transmission rates (such as 1Mbps or 6Mbps) a receiver situated close to the transmitter observes very few frames arriving in error.
Percentage of frames in error at a fixed location.	Increases with the transmission speed and frame size. Decreases with increases in average SNR.	-

4.1.4. Average duration of states for 802.11b and 802.11a systems

The next set of curves present a summary of the results obtained for 802.11 and 802.11a systems. In Figure 32 through Figure 38 the average duration of the good and bad states are plotted for different average SNRs at the receiver, different frame sizes and transmission rates. The averages are the result of computing the mean duration of each state from three separate replications of 100,000 frames each. The confidence levels are not plotted in the figures, but their computation indicated that most of the means are different. In the next chapter an analysis of these curves will be presented along with a discussion of how they can be used to characterize two state Markov models to represent frame losses.

The importance of the following set of curves lies on the fact that this information has not been available in the past. Traditionally, Markov models have been characterized using methods like the one presented in Section 3.3.1.1 to assign durations to each of the states in a FSMC model. This method will be shown to be highly inaccurate for IEEE 802.11 channels in Chapter 5.

On the other hand, the figures presented next allow an assignment based on experimental data. In Chapter 6 we also discuss what issues are relevant when a Markov model is characterized with the experimental data presented throughout this chapter.

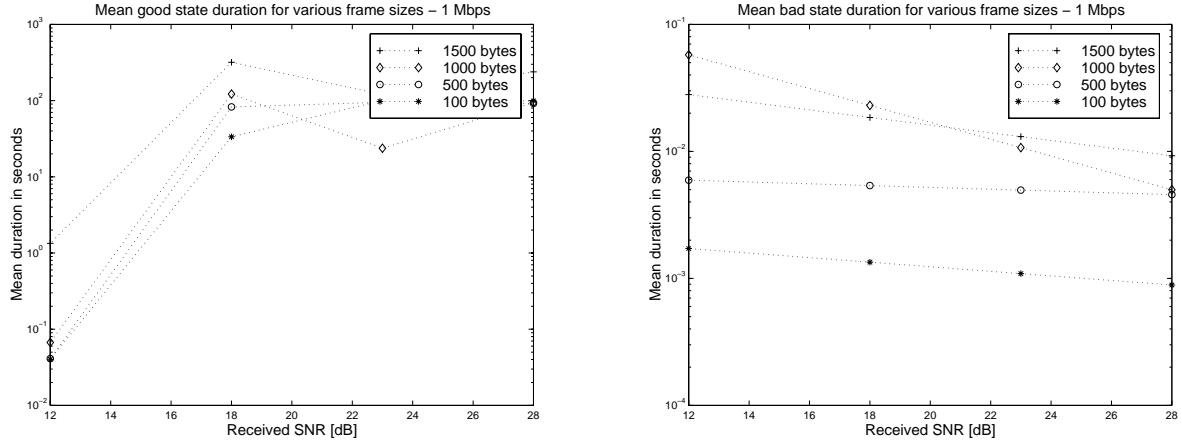


Figure 32 Experimentally observed average good and bad state durations for 1 Mbps transmissions.

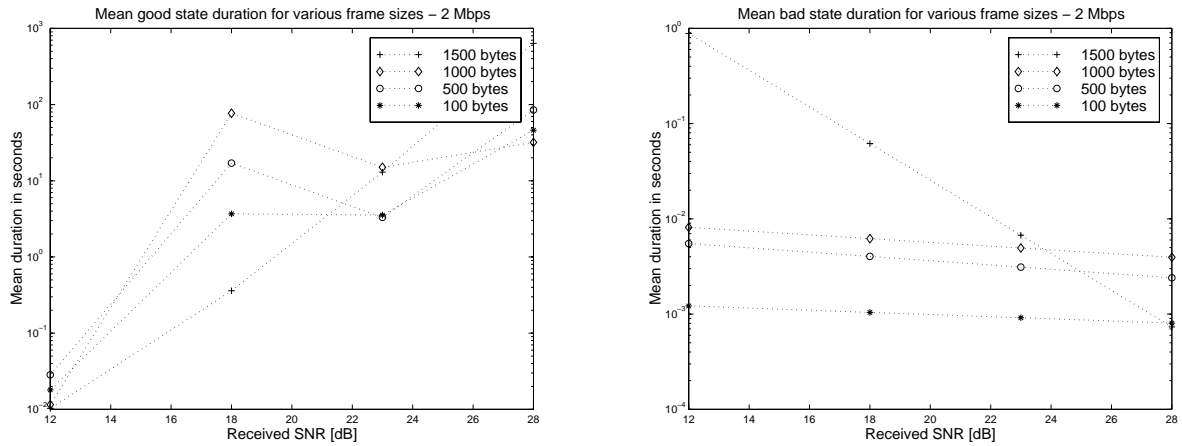


Figure 33 Experimentally observed average good and bad state durations for 2 Mbps transmissions.

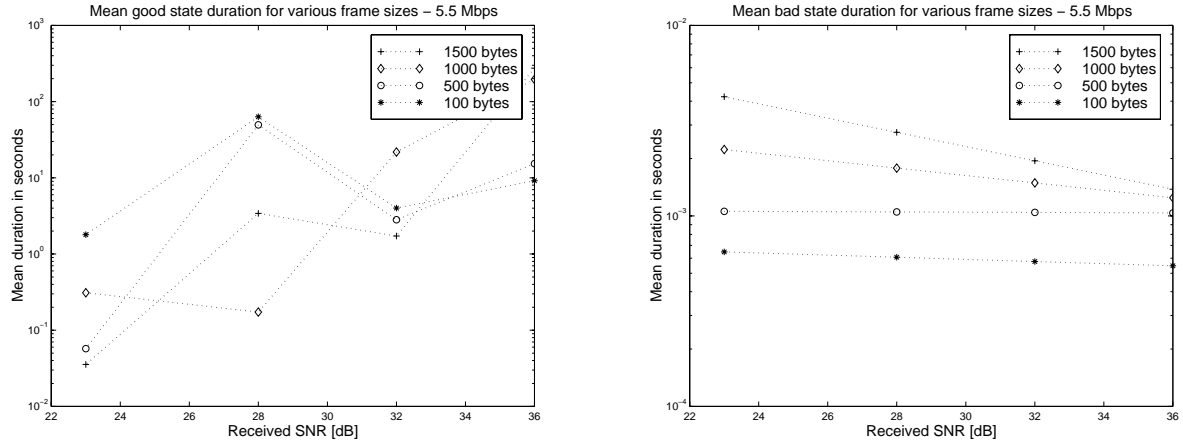


Figure 34 Experimentally observed average good and bad state durations for 5.5 Mbps transmissions.

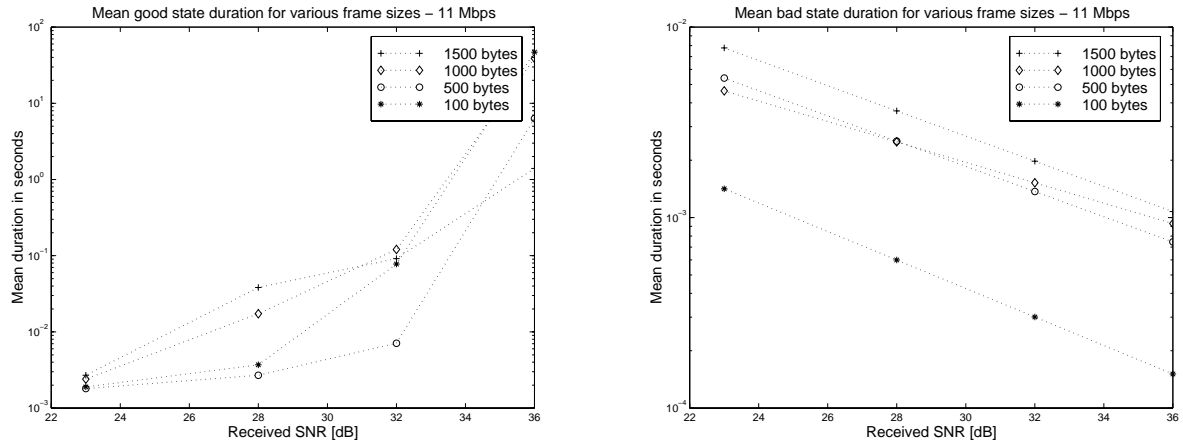


Figure 35 Experimentally observed average good and bad state durations for 11 Mbps transmissions.

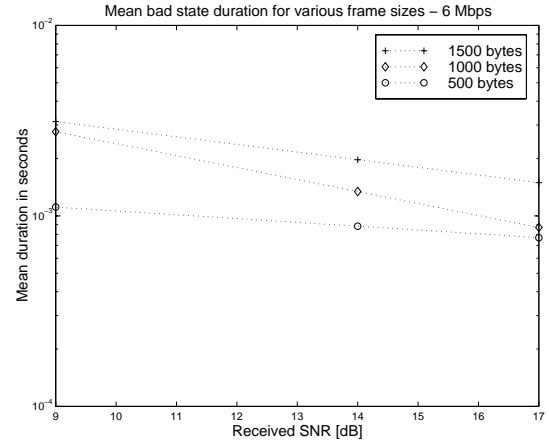
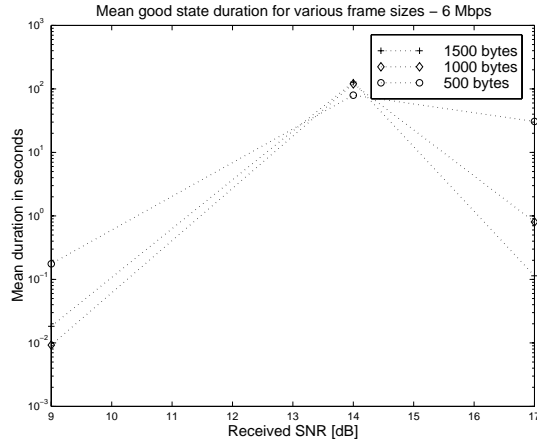


Figure 36 Experimentally observed average good and bad state durations for 6 Mbps transmissions.

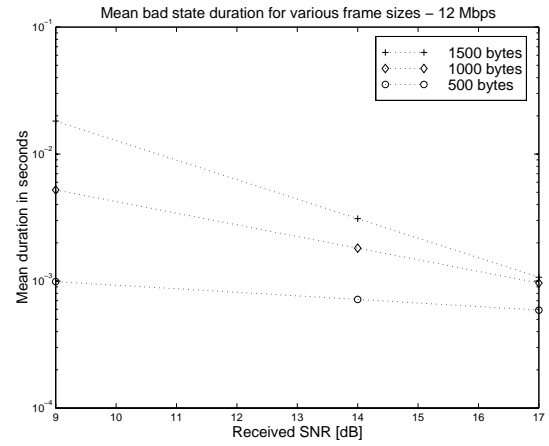
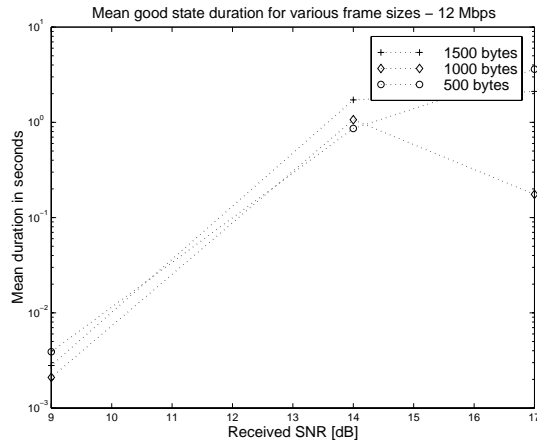


Figure 37 Experimentally observed average good and bad state durations for 12 Mbps transmissions.

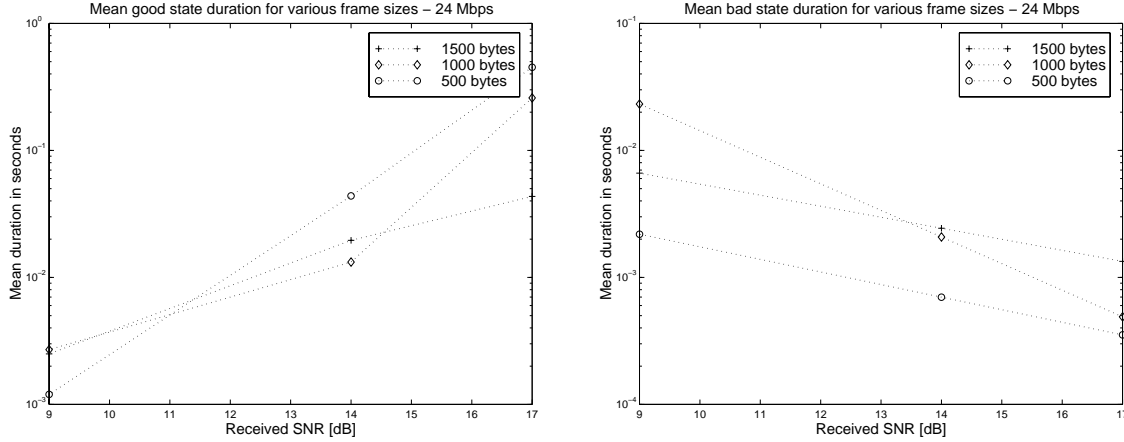


Figure 38 Experimentally observed average good and bad state durations for 24 Mbps transmissions.

From the information presented in Figure 32 through Figure 38 several relevant conclusions can be drawn. For instance the average duration of the good state increases as the average SNR at the receiver increases. These average durations can vary several orders of magnitude with variations in average SNR. For instance, at 1 Mbps and an average SNR of 12dB the observed duration is the range of 0.1 seconds while at 28dB the range is around 100 seconds. This is because the percentages of lost frames under high average SNR conditions rapidly decreases and longer periods of consecutive error free frames are possible. On the other hand, the bad state duration decreases as the average SNR at the receiver increases. These observations are valid for both 802.11 systems.

The average duration of the states is also a function of the frame size. In general with smaller frames the duration of the good state increases since the probability of losing a frame is proportional to the frame size. Furthermore, because of how the data was collected, more accurate state duration measurements are possible with smaller frames. This property of the experiments will be explained in greater detail in the next chapter when guidelines for constructing the models are presented.

The transmission rate and site selection also influence the average duration of each state. When higher transmission rates are used at locations with low average SNR, the duration of the good state can decrease several orders of magnitude. This can be noticed by comparing Figure 32 with Figure 35. At 1 Mbps and 23dB of average SNR the average duration of the good state (for several frame sizes) is between 10 and 100 seconds. On the other hand, at the same location when the transmission rate is 11 Mbps the good state duration is between 0.01 and 0.001 seconds. The same tendencies are observed in both systems, 802.11b and 802.11a.

To characterize a Markov model using Figure 32 through Figure 38 one can directly read from the figure the average value of a state duration. These values can then be assigned to the Markov model that by definition will have geometrically distributed state durations. The selection of a particular value should be done taking into consideration the existing conditions at the site to be modeled. A discussion on the importance of each factor is presented in the next section.

4.1.5. 2^k FACTORIAL DESIGN ANALYSIS OF THE EXPERIMENTAL RESULTS

The interaction between the three factors, average SNR at a location, transmission speed and frame size can be analyzed with a 2^k experimental design (where k represents the number of factors). In this particular case the outputs of interest were the average durations of the error and error free runs. Several combinations of factors were studied. For each case, two transmission speeds were selected and the effect of varying the capture location and frame size on the error and error free run lengths was studied for three replications. For the analysis, the percentage of variation of the outputs was computed as explained by each effect.

In the case of 802.11b systems the conditions analyzed involved the following sets of transmission rates: 11 & 1 Mbps, 11 & 2 Mbps and 11 & 5 Mbps. The selection of these levels was done following the guidelines presented in [24] in which the analysis starts by testing the effect of the minimum and maximum levels of a factor. The selection of locations corresponds to location **a** (highest average SNR observed) and location **d** (lowest average SNR observed) for 1500 and 100 byte frames.

Table 17 shows the percentage of variation of the two outputs explained by the changes in location, transmission rate and frame size in an 802.11b system. The information from the table indicates how for the error free run durations, no factor is predominant except in the 11 and 1 Mbps case in which the transmission rate explains the variations. On the other hand all the error run durations variations depend mainly on the frame size. The larger the frame size the higher the probability of receiving a frame in error and the longer the error runs last.

Table 17 2^k design results for 802.11b experimental data for frame losses.

Effect	11 and 1 Mbps		11 and 2 Mbps		11 and 5 Mbps	
	Error Free Run	Error Run	Error Free Run	Error Run	Error Free Run	Error Run
Tx Rate	55.78	4.46	13.47	0.13	11.00	7.95
Location	3.43	7.41	16.40	12.50	21.30	14.73
Frame size	7.02	63.45	12.84	50.90	9.31	32.29
Tx Rate & Location	0.08	6.26	12.05	11.49	10.67	11.75
Tx Rate & Frame Size	17.07	4.40	17.04	0.41	18.68	6.13
Location & Frame Size	4.04	6.89	12.04	10.62	9.61	13.62
Tx Rate, Frame Size & Location	12.20	5.78	16.12	11.60	19.11	10.90
Experimental Errors	0.39	1.34	0.05	2.36	0.32	2.62

Table 18 shows the percentage of variations of the two outputs in an 802.11a system. 1500 and 500 byte frames were used in locations **a** and **c**. As explained elsewhere 100 byte frames traces generated incoherent information therefore 500 bytes were used for the analysis. Location **d**, was not selected for the analysis in 802.11a since at this location the percentage of lost frames was practically 100% and no useful conclusions can be drawn from such conditions. In 802.11a the variations of the error free run duration are not caused by any predominant effect. The variation of the error run duration is mainly affected by the location and the frame size.

Table 18 2^k design results for 802.11a experimental data.

Effect	24 and 6 Mbps		24 and 12 Mbps	
	Error Free Run	Error Run	Error Free Run	Error Run
Tx Rate	12.78	0.52	20.92	6.00
Location	19.44	35.37	37.87	38.61
Frame size	19.92	27.71	0.66	21.08
Tx Rate & Location	9.27	11.01	20.82	8.18
Tx Rate & Frame Size	11.12	0.05	4.53	0.63
Location & Frame Size	16.23	8.07	0.66	8.88
Tx Rate, Frame Size & Location	8.37	0.84	4.56	0.49
Experimental Errors	2.87	16.43	9.99	16.13

In summary from the two previous tables the following remarks are valuable:

1. In 802.11b and 802.11a systems the average error free run duration does not appear to be affected by any particular factor but by all of them and their combinations.

2. In 802.11b systems the average error run duration is mainly affected by the frame size. Larger frames result in observing higher average durations.
3. In 802.11a systems the average error run duration is mainly affected not only by the frame size but by the location. The differences between the systems lie on the fact they use different frequencies, modulation and coding schemes. The propagation characteristics of the signals and how the channel affects the signals in both systems are therefore different.

4.2. EXPERIMENTAL TRANSMISSION RATE VARIATIONS RESULTS

Using a setup like the one presented in Figure 10 it is possible to obtain experimental traces that contain the information about the transmission rate selected for the exchange of frames between the stations. This section shows results for both 802.11b and 802.11a technologies. The measurements were collected at different locations at office and residential environments.

4.2.1. Rate variations observed with 802.11b technologies

The possible transmission rates with 802.11b technologies vary between 1 and 11 Mbps. However, in practice a subset of the possible rates was observed depending on the characteristics of the measurement location. For example at high SNR conditions, such as those present at location **a** of Figure 9 only 5 and 11 Mbps rates were present. As the receiver moves away from the transmitter other transmission rates are used. At low SNR conditions the lowest rates are usually selected.

Figure 39 illustrates the effect observed in the transmission of 1500 and 1000 byte frames at three different office locations. The figure shows the evolution in time of the transmission rate variations. For clarity in the figure, only 20,000 frames are plotted from 500,000 analyzed in three different replications. Each subfigure also includes the information about the average transmission rate computed from the three different runs. In the next chapter models for these variations will be created by looking at the distribution of the duration of each transmission rate and the average transmission rate.

4.2.2. Rate variations observed with 802.11a technologies

The possible transmission rates with 802.11a technologies vary between 6 and 54 Mbps. As before, in practice only a subset of the possible rates is observed depending on the characteristics of the measurement location. Figure 40 presents similar information to that presented for the 802.11a case.

Using 802.11a technology resulted in more frequent variations in comparison to the 802.11b case; therefore the evolution of transmission rates during only 5,000 frames is shown in each subfigure.

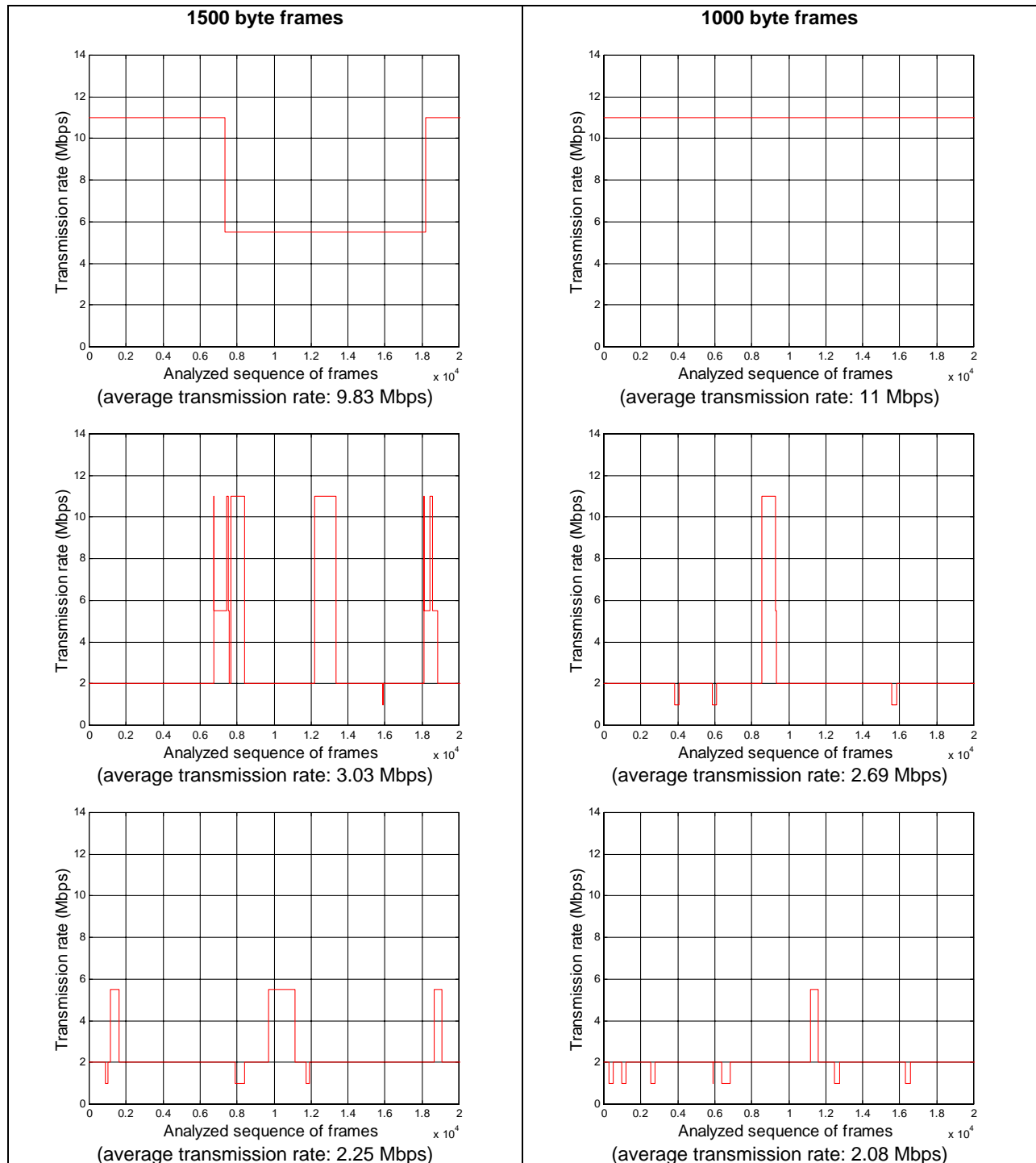


Figure 39 Transmission rates variations observed at different office locations with 802.11b technologies (SNR at the receiver: 36dB (top), 29dB(middle), 27dB(bottom), average rates are shown below each graph)

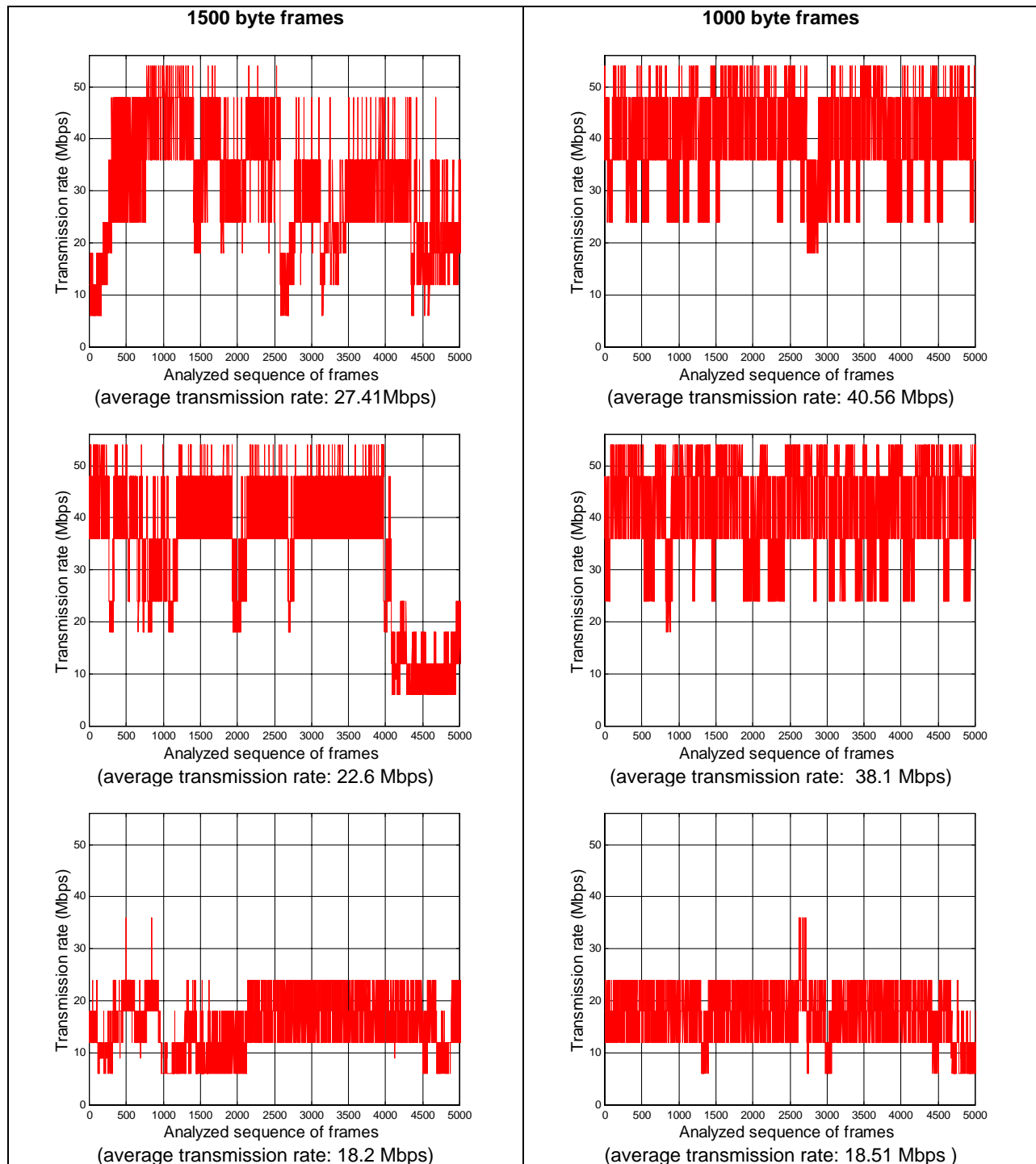


Figure 40 Transmission rates variations observed at 3 different office locations with 802.11a technologies (SNR at the receiver: 17dB (top), 14dB(middle), 9dB(bottom))

For both 802.11b and 802.11a Figure 39 and Figure 40 show how selected rates last for longer periods of time in comparison to the duration of the good and bad states. This can be appreciated by first

looking at Figure 25 through Figure 31 and noticing how the mean duration of the bad state is in the order of tens of frames and hundreds of frames for the good state. On the other hand the duration of each transmission rate is usually in the order of hundreds or thousands of frames per rate. Further examination of this characteristic was performed by looking at the experimental traces and noticing that changes in the RSSI did not necessarily result in immediate changes of the selected transmission rate.

Figure 39 and Figure 40 also show the average transmission rate observed in the experiments. These averages were computed by taking into account three independent traces for each configuration, and computing the mean for each of them. The values shown in the figure are the average of the three independent means. By looking at these values it was noticed how changes in the average SNR at the receiver can noticeably decrease the average rate observed. This was observed in both 802.11b and 802.11a systems. For example, for 802.11b a drop from 9.8 to 2.25 Mbps was observed when the receiving station was moved from a location with 36dB to a location with 27dB of average SNR. In the same figure for the 802.11a case, the average transmission rate dropped from 40Mbps to 18Mbps with a 8dB drop of average SNR.

4.2.3. 2^k FACTORIAL DESIGN ANALYSIS OF THE EXPERIMENTAL RESULTS

The same type of analysis used to understand the importance of each factor on the frame loss process can be applied to the transmission rate variations case. The factors of interest are now the average SNR at a location and the frame size. The output of interest is the average transmission rate. As before, several combinations of factors were studied. For each case, two measurement locations were selected and the effect of varying the frame size was studied for three replications. Again, for the analysis, the percentage of variation of the outputs was computed as explained by each effect.

For both 802.11b and 802.11a systems the conditions analyzed involved one set of locations, location **a** and location **c** with 1500 and 1000 byte frames in an experiment with three replications. The next table shows the results.

Table 19 2^k design results for 802.11b and 802.11a for the transmission rate variations.

	Effect	Location a(highest SNR) and Location c(low SNR)
		Average transmission rate
802.11b system	Location selection	53.1%
	Frame size	35.1%
	Location selection & Frame size	4.5%
	Experimental Errors	7.3%
802.11a system	Location selection	47.5%
	Frame size	28.9
	Location selection & Frame size	14.9%
	Experimental Errors	8.7%

Table 19 shows that for both technologies, the average transmission rate mainly depends on the location selection. This means that the average SNR at the receiver is the main factor that influences the mean rate observed during the transfer of frames. This effect is also noticeable in Figure 39 and Figure 40 in which decreases in SNR result in lower average transmission rates.

5. LIMITATIONS OF EXISTING MODELS

Chapter two and three provided the theoretical fundamentals and the details for creating and validating models for frame losses that occur in wireless channels. In order to understand the motivation behind the construction of models based on experimental data it is relevant to illustrate the limitations of current approaches. These limitations foster the need for models that can accurately represent frame losses. Traditional models use factors that do not have a direct relationship with some common channel variables or assume simplified operating conditions. Without any direct quantitative link between the model characterization and the channel itself it is very complicated to assign values to the model parameters. At best, the procedure will give inaccurate results. In this chapter we will explore the limitations and illustrate the complexities of characterizing a Markov model.

5.1. LIMITATIONS IN THE CHARACTERIZATION OF THE MODELS

The limitations that exist in the traditional characterization of models like the FSMC make it difficult to assign with certainty, values to the duration of the states. In summary the limitations that one faces in this process are summarized in Table 20.

Table 20 Limitations in the characterization of traditional models

Model	Limitation description	Importance
FSMC and HMM	Characterization based on the characteristics of the fading envelope has been only developed for narrowband channels that use simple modulation schemes.	IEEE 802.11 technologies use wideband channels with complex modulation schemes.
FSMC	<p>There is no quantitative method to accurately tune the duration of each state of the model to factors such as:</p> <ul style="list-style-type: none"> • Transmission rate • Frame size • SNR at the receiver • Type of environment (residential or office) <p>For complex technologies like 802.11 it is quite difficult and time consuming to develop quantitative means to tune the states' durations.</p>	<p>These factors are quite relevant in any performance study. In particular:</p> <ul style="list-style-type: none"> • Transmission rate determines the transfer times of frames and varies according to the condition of the channel. Models are usually constructed assuming the transmission rate stays constant during operation; this is usually not the case in IEEE 802.11 systems. • The probability of receiving a frame in error varies with the frame size. • The SNR is a very common side parameter used to represent the quality of the channel. Usually a site is mainly characterized by its SNR. Survey tools commonly use it to report status of the channel. • The propagation characteristics of each environment vary according to the construction materials.

5.1.1. Effects of parameter selection in Markov models

When there is no experimental data to base the model on, characterizing a Markov model *so that its output matches the characteristics of a frame loss process* could be quite complex. To illustrate this complexity let's assume that the frame loss process can be characterized using a four state Markov model. The characterization of this model is typically done based on a simulated received signal envelope generated with the procedures mentioned in Appendix C.

First, the partitioning of a simulated SNR received envelope can be done with one of the schemes described in Appendix A. For this example the second scheme, defined by equation (2) was selected. As shown in [2] this scheme gives a better fit in terms of the values of the steady state vector π . However it is important to mention that no particular partitioning scheme would result in any major improvements in terms of the model's parameters, as shown in [2]. For this case a carrier frequency of 2.4 GHz (the same as that one used in 802.11b systems) and a walking speed of 2 Km/hr can be selected for the generation of the simulated envelope. Figure 41 illustrates how the simulated received envelope was partitioned to create four intervals that correspond to the states of the model.

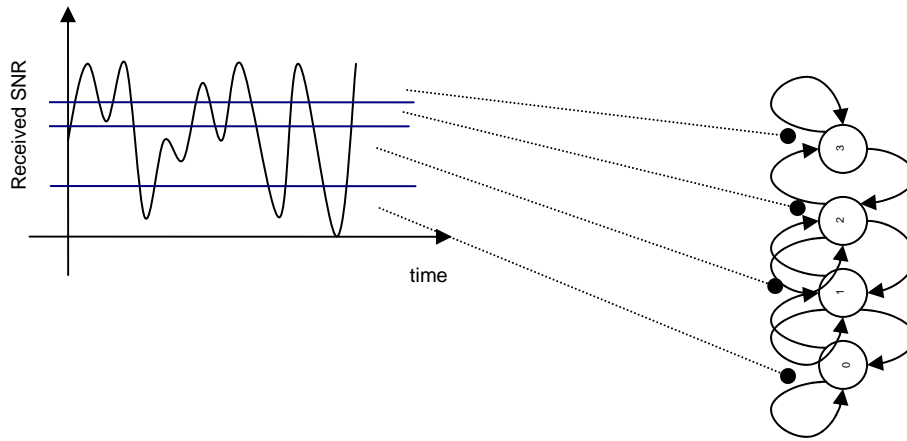


Figure 41 SNR envelope partitioning and state assignment to a four state FSMC model.

By analyzing the simulated envelope and counting the number of times it traverses between states it is possible to obtain the values for the vector π and the matrix \mathbf{P} . It is relevant to mention that actual values of received average SNR are not taken into consideration in the generation of the envelope.

After obtaining the values for π and \mathbf{P} , it is necessary to determine how frames will be discarded or accepted by the model. Three configurations were tested for this particular case. For each configuration *a different percentage of frame loss* was assigned to each state for each run. The runs consisted of transmissions of 100,000 frames. These values are illustrated in Table 21.

Table 21 Configuration settings for a four state FSMC

State	Percentage of lost frames		
	Configuration 1	Configuration 2	Configuration 3
0	100	100	100
1	70	80	90
2	30	40	50
3	0	0	0

Since the percentage of lost frames in each state is not known when the model is characterized, constructing it is challenging. Only a few studies [53][57] have related the frame size and simple modulation schemes to the loss process, but *only with simulated envelopes* and for narrowband channels. This characterization method provides no formal way of assigning the percentages shown in Table 21 and so the selection of the percentages was done in a trivial way.

The only guideline followed for assigning the percentages is that for higher SNR conditions a lower number of frames should be lost. In order to evaluate how well the frame loss process generated by this model performs with such percentages, an OPNET simulation was implemented. In the simulation an 802.11 environment was constructed for the transmission of 1500 byte frames at 11Mbps between a transmitter and a receiver.

Figure 42 shows the simulation setup used for generating the simulated traces. As with the experimental traces, these are sequences of zeros and ones that represent the state of the channel. These sequences are generated at the receiver which includes the four state FSMC model. The setup shown in the figure is similar to that illustrated in Chapter 3.

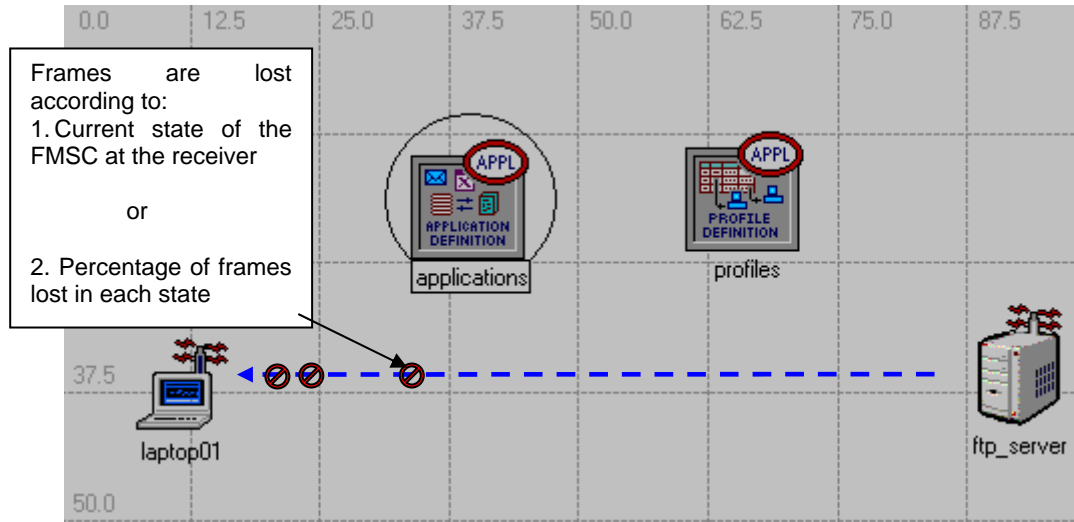


Figure 42 Simulation setup for comparing the frame loss traces effects on an FTP file transfer.

The traces obtained via simulation can be directly compared to similar experimental scenarios. With this in mind, the next figure shows the histograms of the error and error free runs obtained with the

setup from Figure 42. Only one result is shown; however, the histograms from the three configurations had similar ‘exponential’ like shapes.

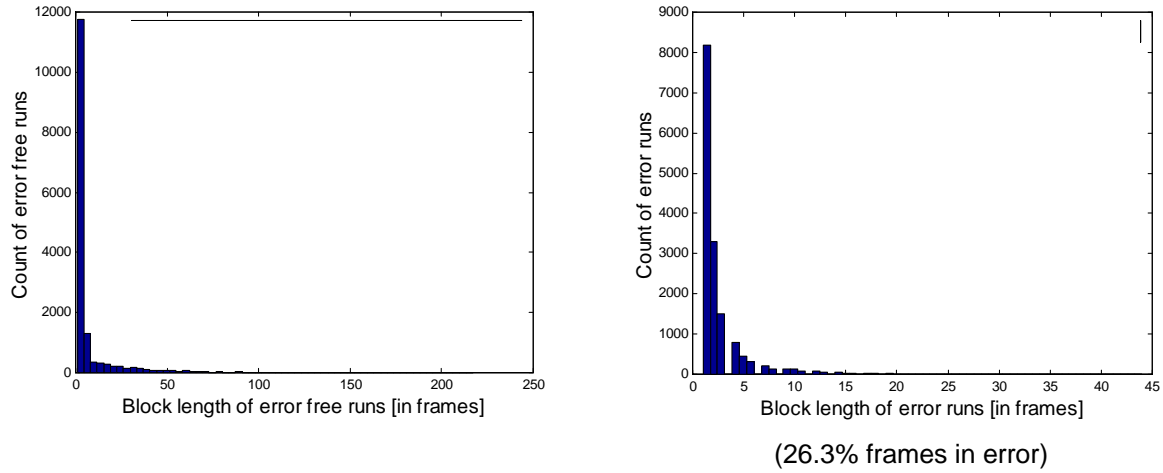


Figure 43 Histograms from the simulation using the setup from Figure 42 with a 4 state FSMC.
(tx rate=11 Mbps, frame size=1500 bytes, percentage of frames lost per state: 100%, 70%, 30% and 0%)

In order to understand if any of the configurations tested approached the observed characteristics of the frame loss process one can compare a second variable, the observed percentage of frames received in error. This percentage can be computed by looking at the total number of frames received in error in each run and then computing the average for three runs. The results, for the three configurations along with the actual average observed results at four different SNR at the receiver are plotted in the next figure.

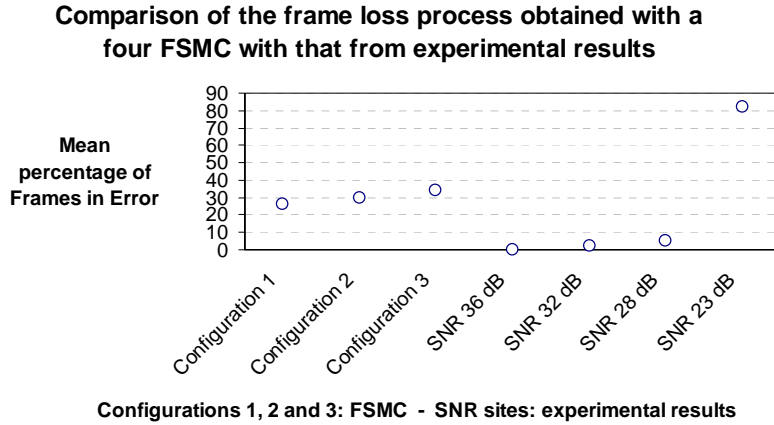


Figure 44 Comparison of results for a four state FSMC.

(The y axis percentage is the mean percentage from three runs of 100,000 frames each)

Figure 44 clearly shows how the results of using a four state FSMC differ significantly from those observed in the experiments. The mean percentage of frames in error from the four-state model is quite different from the experimental one. This is due to the fact that for all the configurations tested the configuration of the FSMC model did not generate a ratio of bad state to good state duration that is similar to the one from the experiments.

It may be possible to tune the parameters of the FSMC to obtain a similar output to the one observed in the experiments by taking into account other elements. However, incorporating elements, such as the modulation or coding scheme, in the characterization process cannot be done directly. The complexities of modulation and coding in 802.11 and 802.11a systems cannot be directly related to the loss process of the model. This means that trying to relate them with the frame loss percentages selected for each state does not give any additional information.

In the same manner increasing the number of states does not solve the characterization inaccuracies. For example, in Chapter 4 two-state models were constructed with information from the fading envelope and matching the experimental observations was again impossible. Four-state models were used in this chapter and the same problems were encountered. There is no quantitative way of determining an adequate number of states or how to assign the loss process to each state.

Another approach that could be considered is the construction of a Markov model that includes the effects of frequency selective fading. Such a model has only been proposed in [27] but no accuracy or

validity tests have been performed. Furthermore currently there are no guidelines that indicate how the model can be constructed.

Incorporating elements such as the number of states, partitioning and fading process in the characterization of a frequency selective fading would result problematic. This is due to the fact that there is no direct way of relating the observed frame loss percentages to the model's elements.

6. MODELING OF THE CHANNEL AND MODEL VALIDATION

6.1. MODEL RESULTS AND VALIDATION OF THE FRAME LOSS PROCESSES

The experimental data collected at the different sites was used to create numerous models. In this section the results obtained from the characterization of frame losses will be presented. The validity of each model will also be studied with the mathematical and simulation tools described in chapter 3. In this chapter we first study the validity of the traditional two-state FSMC model for frame losses when characterized with data coming from a fading envelope. Then the validity of using experimental data for the characterization is analyzed for both the FSMC and hidden Markov models. In Section 6.1.3 both of these models are also used to study the effects of frame losses on upper layers. These effects are studied by looking at the transfer time of files using TCP as the transport protocol.

6.1.1. FSMC model

6.1.1.1. Findings from the characterization based on the attributes of the fading envelope

Using the equations developed by Jakes [23] and the procedure detailed in Chapter 3 it is possible to assign values to the state durations of a two state Markov model. Following that procedure several models were constructed for various fade depths values ranging from 5dB to 70dB below the r.m.s value of the received envelope.

Notice that in the procedure from Chapter 3, for the computation of the states' durations a non-zero Doppler frequency value is required. In the actual experiments the stations were stationary, then a mobile velocity of 0.001 m/s was assumed for the Doppler frequency calculation. This velocity not only allows the computation of a non-zero Doppler frequency but also allows the inclusion of changes in the environment between the transmitter and receiver. In an environment such as an office or a residence changes occur slowly when pedestrian speeds are considered.

For each fade depth, results were generated and compared to the data from the experimental traces. It was found that none of the configurations adequately approximated the observed frame loss sequence characteristics for any of the arrangements tested. Even though larger fade depths gave

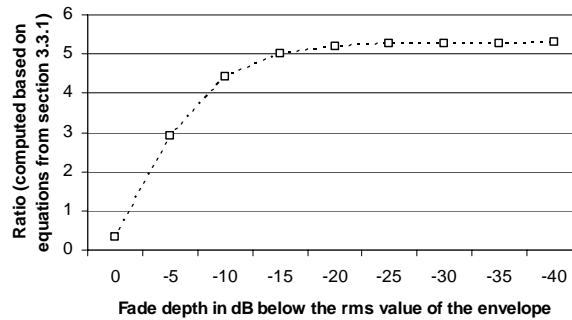
apparently better results, neither the distributions nor the frame loss percentage were similar to the experimental results. This is mainly due to the fact that the method from Chapter 3 does not generate a ratio between the good states and bad state durations that approximates any of the experimentally observed ratios.

By analyzing the received frame sequences, the lengths of the error free and error runs between the experimental and the simulation results were compared. The results showed significant differences. The differences were typically in the order of 1000%. This clearly indicated that the method from Chapter 3 does not give adequate results for computing the average state duration when 802.11 channels are being characterized with two-state Markov models. This can be easily visualized in Figure 45. The figure shows the ratios between the average duration of the error free periods and the average duration of the error periods. In Figure 45 the results obtained with the method detailed in Section 3.3.1.1 is plotted in the top subfigure. The experimental results are plotted on the second and third subfigures.

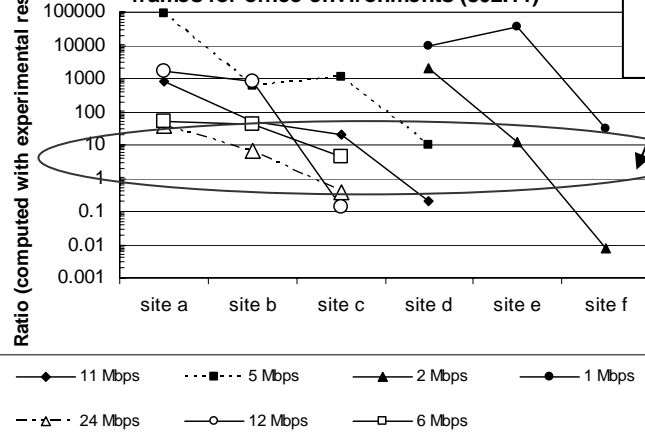
Figure 45 shows how the range of ratios that can be obtained from Jakes equations varies between 0.3 and 5 (top subfigure). On the other hand, the range of experimentally observed ratios varied between 100,000 and 0.001 for both 802.11b and 802.11a systems. This means that assigning durations to each of the states of the two-state Markov model, is only possible in a very small region if the equations proposed by Jakes are used. Additionally, notice that with Jakes equations there is no direct mechanism to link the expressions to an average value of received SNR. This suggests that the assignment of durations should be done in a different way. The mechanism to effectively assign accurate durations to the states will be discussed in a later section of this chapter.

Figure 45 only serves as a reference for comparing theoretical and experimental results. It can not be used to assign state durations. It is necessary to know at least the duration of one state to compute the duration of the other if the ratio of the two is known in advance. More information is needed to characterize the channel; an alternate process will be proposed later in this chapter.

Average error free run duration to average error run duration ratios for several fade depths



Average error free run duration to average error run duration ratios - in a transfer using 1500 byte frames for office environments (802.11)



Average error free run duration to average error run duration ratios - in a transfer using 1500 byte frames for residential environments (802.11a)

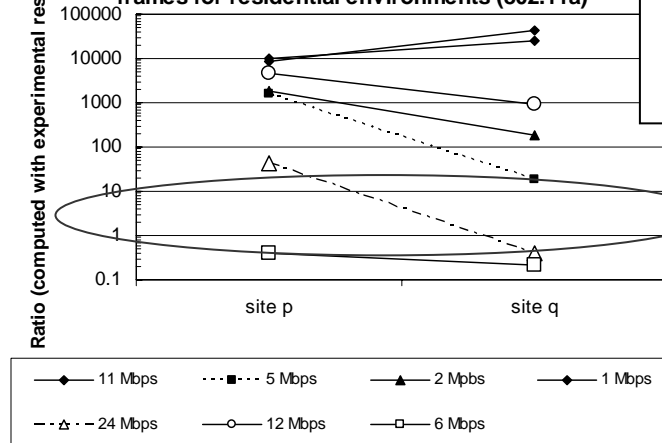


Figure 45 Theoretical and experimentally observed average state duration ratios.

6.1.1.2. Findings from the characterization based on experimental data

A practical approach for the assignment of states' durations is the usage of the set of curves presented in Section 4.1.4. Those curves were generated with the experimental data collected in an office environment. The plots allow the assignment of durations to both states of a two state Markov model for 802.11b or 802.11a systems under different transmission rates, frame sizes and average SNR at the receiver. In the curves, the average durations of the error free and error runs are referred to as average good state and bad state durations respectively.

The next figure shows a sample of the histograms coming from a two-state Markov model constructed using Figure 35. In this particular case, the results shown correspond to the modeling of a transmission of 500 byte frames at 11 Mbps at location **d**. As expected, the histogram follows a shape that closely resembles an exponential distribution. The left subfigure shows the histogram of the error free runs, the right one presents the one for the error runs.

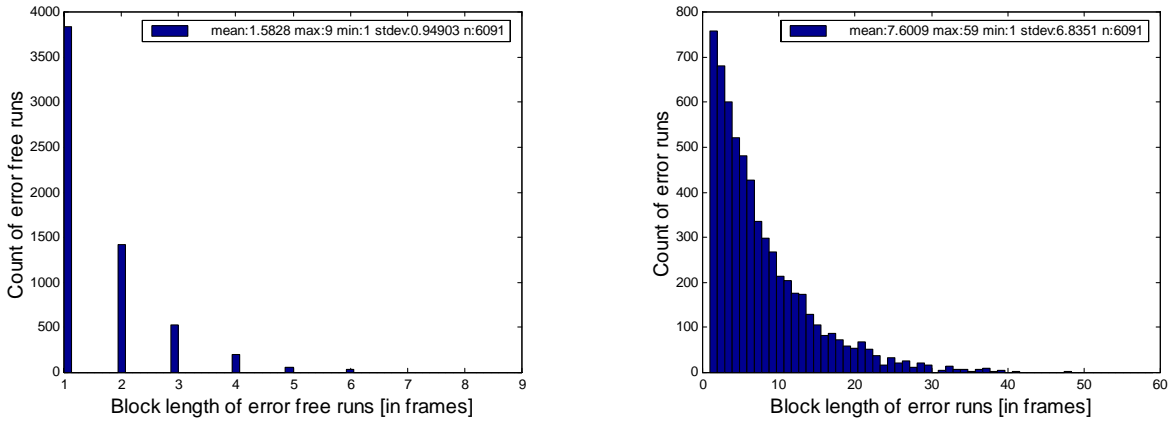


Figure 46 Error run and error free run distributions generated with a two-state Markov model.

(The characterization was done using data taken from Figure 35 with 500 byte frames at a SNR of 23dB)

A first comparison of these results with the experimental traces can be done by looking at the percentage of frames received in error. This variable provides a comparison basis for the durations of both the good and bad states. Table 22 shows a sample of the results from this comparison for several configurations including that of Figure 46. For Table 22 transmission rates of 11, 5.5 and 24 Mbps had been selected for the computation of the percentage of frames in error at a SNR of 23dB.

Table 22 Comparison of the simulation results vs. experiments for a two-state FSMC

Transmission rate [Mbps] - frame size [bytes]	Two State Model Parameters for the simulation (from Fig. 32-38)		Average % of Frames in error from a simulation	Average % of Frames in error from the experimental traces
	Mean good state duration [s]	Mean bad state duration [s]		
11 – 1000	0.0173	0.0015	7.98%	9.54%
11 – 500	0.0027	0.0037	57.79%	56.65%
5.5 - 1000	0.1730	0.0023	1.35%	1.39%
5.5 – 500	49.514	0.0007	0.002%	0.01%
24 – 1000	0.0132	0.0015	10.12%	10.15%
24 – 500	0.0438	0.0004	0.88%	0.89%

Table 22 shows the mean durations of states assigned to a model in six configurations each with diverse transmission rates and frame sizes. It also contains the average percentage of frames in error observed in three replications of the simulation and from the experimental traces. From these results it can be seen that the differences in the percentage of frames received in error between the simulation results and those observed experimentally are not significant. However, matching this percentage does not necessarily mean that the modeled distributions of error runs match the ones from the experiments. The underlying modeled distributions could have completely different characteristics and the percentage could still be similar.

To compare the distributions of the error and error free runs the Kolmogorov-Smirnov (K-S) test for two independent samples (see Chapter 3) can be used. This test will indicate if the differences in the distribution shapes are significant or not. Following the guidelines give in [30] a Z test asymptotic value of 0.05 is generally used to decide if these differences are significant or not. Values below 0.05 indicate significant disparities between the distributions, values above 0.05 indicate similar distribution shapes.

The results from the two-sample K-S tests are shown in Table 23. In this particular case, the results correspond to 1000 byte frames transmitted at different rates with the conditions existing at location **d** (for 802.11b rates) and location **c** (for 802.11a rates). The table shows for each configuration the results of the test for both the good and bad states distributions. The asymptotic value columns indicate if the shapes of the distributions from the model are similar or not to those observed in the experiments.

Table 23 Sample K-S test results for the comparison of the states' distributions

Transmission rate [Mbps] - frame size [bytes]	Good state distribution	Bad state distribution
	K-S test shape result	K-S test shape result
11 – 1000	>0.05	<<0.05
5.5 – 1000	>0.05	<<0.05
2 – 1000	0.161	1.001
1 – 1000	.95	1.000
24 – 1000	<0.05	<0.05
12- 1000	<0.05	<0.05
6 – 1000	<0.05	<0.05

Notice that for the lower transmission rates of 1 and 2 Mbps the asymptotic value indicates these distributions are similar. For the rest of the cases the distributions have different shapes. This is due to the fact that at low transmission rates, the percentage of frames in error is lower than 0.1% (see Figure 26 for 1 and 2 Mbps) and matching non-geometric like distributions with Markov models is easy since errors occur just sporadically. This happens because under low rate conditions the model only needs to generate very long error free periods followed by short and sporadic error periods.

On the other hand, when the percentage of frames in error is significant, which is usually the observed “average” case, then the shapes resemble a geometric distribution. However as shown in Table 23, the K-S test shape results have values lower than 0.05 and therefore the majority of the experimental distributions have significant differences with the ones generated from the model. This was observed for all the configurations tested in both 802.11b and 802.11a office and residential environments. Therefore a better approach, like using hidden Markov models, was tested.

6.1.2. HMM

The characterization of hidden Markov models to represent frame losses in 802.11b and 802.11a systems was done following the steps illustrated in Figure 17. The experimental traces contained the information needed to obtain a first model that was later used to run the Baum Welch algorithm.

6.1.2.1. Findings from the characterization based on experimental data

In order to make a first comparison of the HMM's output with the experimental results, histograms of the error free and error runs were constructed. Additionally, Kolmogorov-Smirnov tests for

two independent samples were used to quantify the differences between the distributions. Figure 47 shows a comparison between the experimental distributions and those coming from the HMMs. The histograms in the figure correspond to the transmission of 1500 byte frames at 11Mbps in an office environment at two locations, **c** and **d**.

For Figure 47 the scale of each HMM result is chosen to be the same as the corresponding experimental subfigure. This helps in the visual comparison of the histogram results. By visually analyzing the figure it is clear that the distributions generated with the HMMs do not possess those very long tails commonly found in the experimental results. For example at 20dB, the experimental error free run has a tail with a period 2,169 consecutive frames, while the HMM is able to generate an output with a much shorter tail of 129 frames.

These differences in the tail sizes result in differences in the percentage of frames in error between the experimental results and the modeled ones. These differences at location **c** are only of 0.2%, but at location **d** the difference is in the order of 8%. However by looking at the traces it was noticed that these differences are created by samples that contribute with less than 1% to the cumulative frequency of the error run distribution.

Similar results were observed in other 802.11b and 802.11a configurations for both office and residential environments. A sample of the results for an 802.11a configuration is shown in Figure 48. For the figure a transmission at 24Mbps with 1500 byte frames at location **d** was selected. Notice how the tails generated by the HMM in Figure 48 are also shorter than those observed in the experiments.

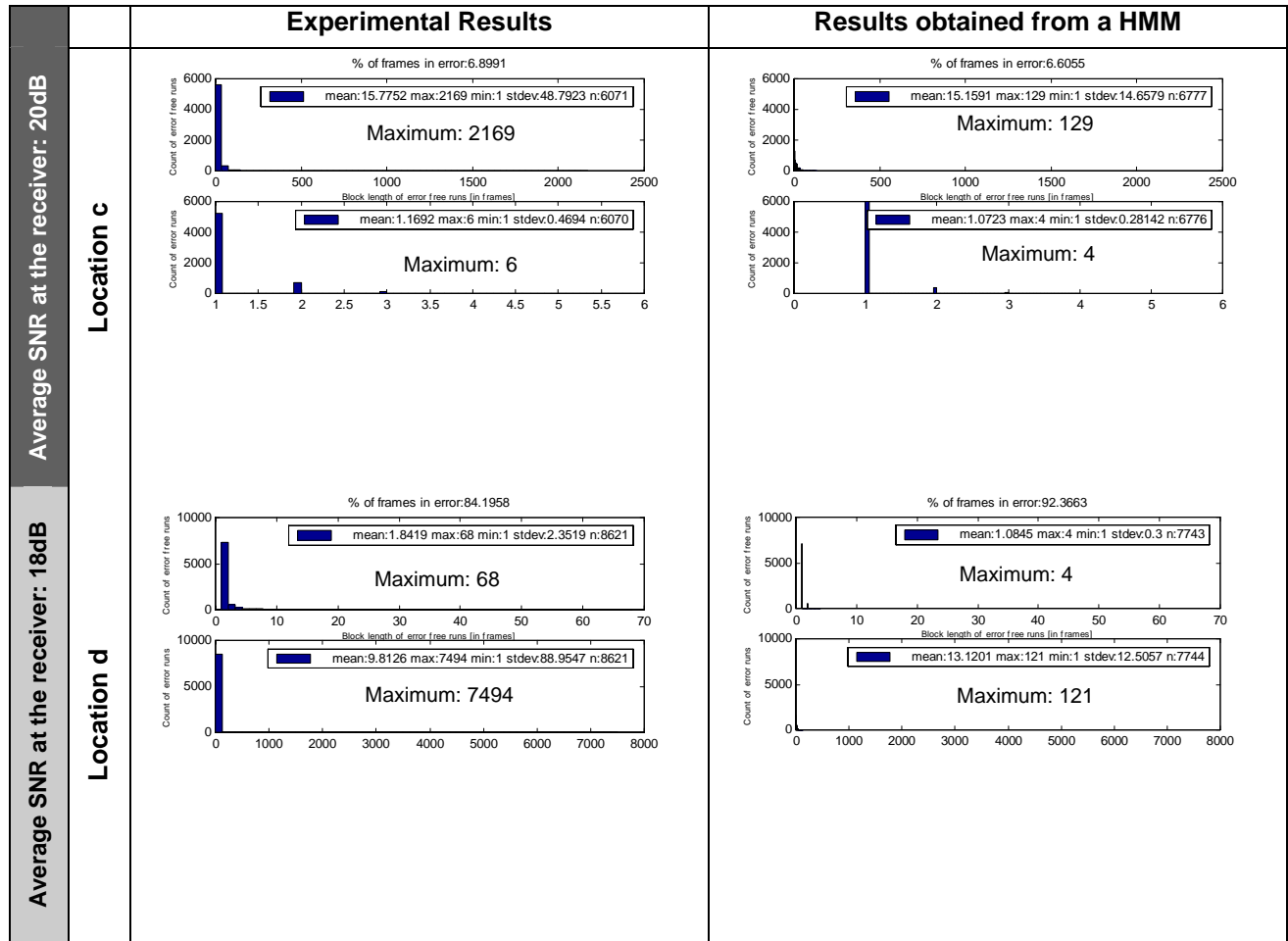


Figure 47 A sample of experimentally and HMM histograms in the office environment at two locations. (802.11b, average SNR at location **c**=20dB, average SNR at location **d**=18 dB, frame size=1500 bytes, tx rate = 11 Mbps)

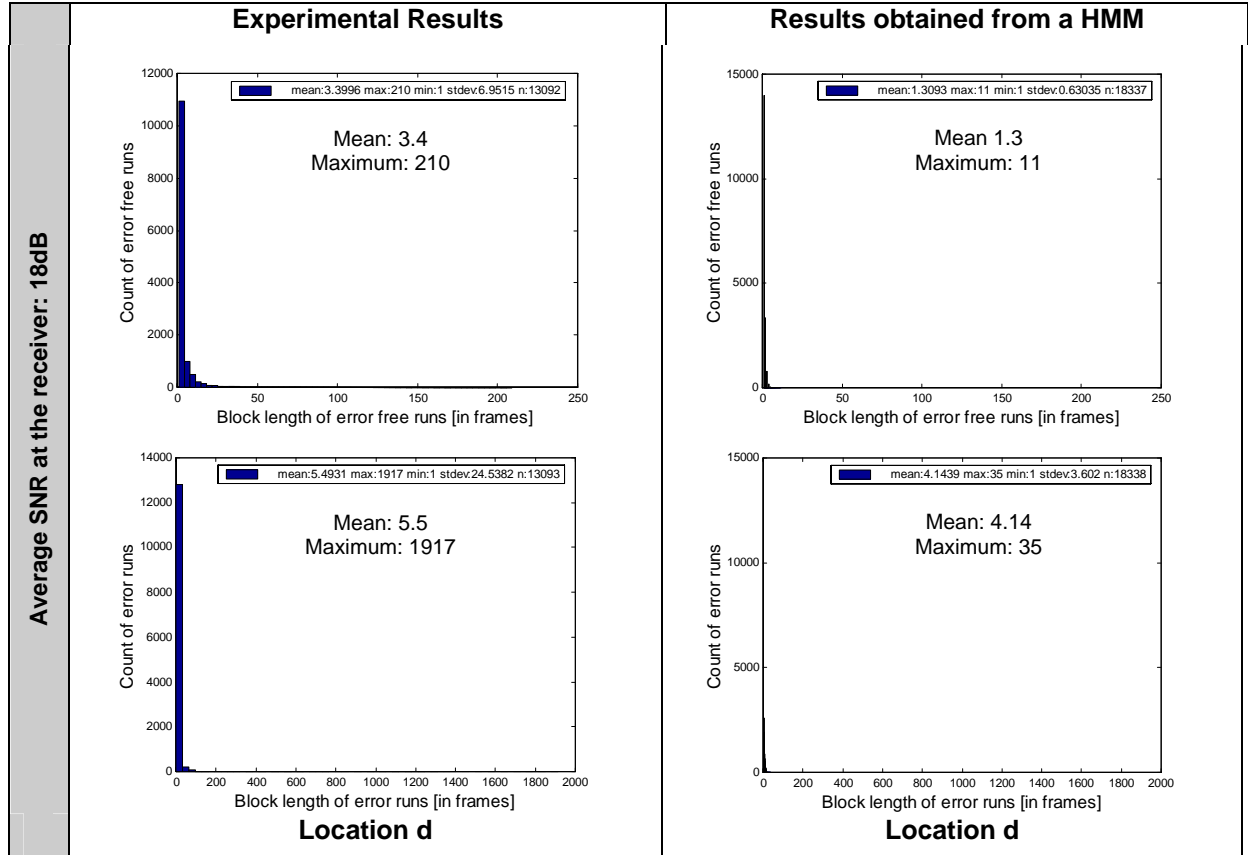


Figure 48 A sample of experimentally and HMM histograms in the office environment.
(802.11a, average SNR at location **d**=18 dB, frame size=1500 bytes, tx rate = 24 Mbps)

In order to quantify the differences between the HMM and the experimental distributions a two-sample K-S test was used. For this test, since the experimental distributions have long tails that are generated by a very low number of samples, these samples were considered outliers. The criteria suggested in [24] for identifying these outliers was used before the K-S tests were computed. In particular, this was done by first ordering the samples in ascending order and then computing the cumulative percentage of each sample. Tail samples with a cumulative percentage above 99% were considered outliers and not taken into account. In order to compare the shape of the two distributions, the experimental and the HMM results were centered around zero on the horizontal axis before computing their similarity with the K-S test.

The following table summarizes the results obtained from the K-S tests in an 802.11b office environment. For different frame sizes and locations the table shows the means and the results for the similarities in distribution shapes. In particular, whenever the resulting asymptotic value is more than 0.05 it is considered that the two distributions have similar shapes.

Table 24 K-S test results for the comparison between HMM and experiments for 802.11b systems

Configuration	Error free runs			Error runs		
	Mean (Exper. results)	Mean (HMM results)	K-S Test Shape result	Mean (Exper. results)	Mean (HMM results)	K-S Test Shape result
1 Mbps						
1500 bytes, location e	5600	5710	>0.05	100000		>0.05
1500 bytes, location f	109.8	145.2	>0.05	1.3	1	0.001
1000 bytes, location e	12078	12312	>0.05	1	1	>0.05
1000 bytes, location f	11001	11231	>0.05	1.5	1.4	>0.05
2 Mbps						
1500 bytes, location d	1976	2520	>0.05	1	1	>0.05
1500 bytes, location e	10300	13501	>0.05	7.5	7	>0.05
1000 bytes, location d	3200	3301	0.01	1.13	1	>0.05
1000 bytes, location e	16161	16606	>0.05	1	1	>0.05
5.5 Mbps						
1500 bytes, location c	137	153	0.01	1.32	1.00	>0.05
1500 bytes, location d	9.5	8.1	>0.05	1.3	1.15	>0.05
1000 bytes, location c	94.2	225	0.001	1.2	1	>0.05
1000 bytes, location d	16700	3300	0.001	1	1	>0.05
11 Mbps						
1500 bytes, location c	15.7	14.9	>0.05	1.17	1.09	>0.05
1500 bytes, location d	1.8	1.83	>0.05	9.8	13	0.001
1000 bytes, location c	10900	10000	>0.05	1	1	0.001
1000 bytes, location d	2753	2700	>0.05	1	1	>0.05

In a similar way for the office environment corresponding to 802.11a a summary of the results is presented in the next table.

Table 25 K-S test results for the comparison between HMM and experiments for 802.11a systems

Configuration	Error free runs			Error runs		
	Mean (Exper. results)	Mean (HMM results)	K-S Test Shape result	Mean (Exper. results)	Mean (HMM results)	K-S Test Shape result
6 Mbps						
1500 bytes, location c	6.79	4.74	>0.05	1.85	1.27	>0.05
1500 bytes, location d	17.2	16.1	0.001	2.75	1	0.002
1000 bytes, location c	7.01	3.9	>0.05	2.2	2.3	>0.05
1000 bytes, location d	16.2	15	>0.05	1.6	1.01	0.01
12 Mbps						
1500 bytes, location c	1.86	1.04	0.001	33.2	21.1	0.001
1500 bytes, location d	7.8	6.9	>0.05	1.6	1.2	>0.05
1000 bytes, location c	2.4	2.25	>0.05	7.4	6.8	>0.05
1000 bytes, location d	6.7	6.5	>0.05	2.03	1.47	0.001
24 Mbps						
1500 bytes, location c	3.4	3.3	>0.05	5.4	4.1	0.001
1500 bytes, location d	13.2	12.7	>0.05	9.17	9.1	>0.05
1000 bytes, location c	4.3	4.01	>0.05	3.5	3.5	>0.05
1000 bytes, location d	2.6	2.3	>0.05	39.1	31	0.01

Table 24 and Table 25 show that in most cases the state duration distributions obtained for the model have similar shapes to those observed in the experiments. This definitely suggests that using hidden Markov models for the frame loss process results in a better representation of the channel than using two-state Markov models.

6.1.3. Findings from the validation by looking at the effects on upper layers

A two-state Markov model characterized with experimental data, a hidden Markov model and a trace driven frame loss model were implemented in a computer simulation like the one illustrated in Figure 23. This was done to study the effects of each model on upper layer protocols. The simulation was configured to transmit 10MByte files between the sender and the receiver using FTP over TCP (Tahoe). For each transmission rate available in an 802.11b system, each file was transmitted three times and the total average transfer time was computed. The average transfer times from the Markov and HMM were compared against the time taken by the trace driven simulation. The procedure was repeated for every SNR ratio data available from the traces. These SNR values correspond to the office locations **a** through **d**. The results are presented in Table 26.

Table 26 Comparison of the average transfer time of a file using FTP and several frame loss processes
(n/a means the file was not transferred due to frequent TCP disconnections)

11 Mbps						
SNR (dB)	FSMC		Trace		HMM	
	time (sec)	file size	time (sec)	file size	time (sec)	file size
23	1175	100000	n/a	100000	n/a	100000
28	14	10000000	15	10000000	14.5	10000000
32	13.1	10000000	13	10000000	13	10000000
36	13	10000000	13	10000000	13	10000000

5.5 Mbps						
SNR (dB)	FSMC		Trace		HMM	
	time (sec)	file size	time (sec)	file size	time (sec)	file size
23	26	10000000	23	10000000	28	10000000
28	20	10000000	20	10000000	20	10000000
32	19	10000000	20	10000000	21	10000000
36	19	10000000	19.5	10000000	20	10000000

2 Mbps						
SNR (dB)	FSMC		Trace		HMM	
	time (sec)	file size	time (sec)	file size	time (sec)	file size
15	n/a		n/a		n/a	
19	65	10000000	66	10000000	67	10000000
23	47	10000000	48	10000000	47.3	10000000

1 Mbps						
SNR (dB)	FSMC		Trace		HMM	
	time (sec)	file size	time (sec)	file size	time (sec)	file size
15	100	10000000	93	10000000	93	10000000
19	92	10000000	90	10000000	90	10000000
23	91	10000000	90	10000000	90	10000000

In Table 26 the column labeled “trace” displays the average transfer time of the file when the loss model is driven by the actual experimental data. The table shows that the average transmission times of the files when using the loss model generated by the hidden Markov model results in a better approximation of the observed results. Nevertheless, the transfer times obtained with the FSMC also show acceptable average transfer times. This clearly indicates that at the transport layer both the two-state and the hidden Markov model provide an accurate representation of reality.

6.1.4. Summary of findings and frame loss processes modeling guidelines

In this section we will elaborate on the relevant findings determined after the experimental models have been validated. Additionally, a set of suggested guidelines for the characterization of frame losses in 802.11b and 802.11a channels will be developed.

- The characterization of a two-state Markov channel that represents frame losses in 802.11b or 802.11a channels is not accurate when done with the equations developed by Jakes [23]. These equations are valid only in a small range of operation and there is no quantitative way of linking them with factors like average SNR, frame size or transmission rate.
- It is possible to use experimental data from experimental traces to characterize two-state Markov models. However the shape of the distributions of errors and no errors periods from the model will not accurately resemble the ones observed in the experiments. Hidden Markov model generate error and error free distributions that closely resemble those observed experimentally.

6.1.4.1. Model construction guidelines for frame loss processes

There are several approaches that can be taken to construct frame loss models. Two-state Markov models or hidden Markov models can be constructed for representing frame losses in 802.11 systems. Both approaches provide good approximations for the percentages of frame in error; however, only hidden Markov models generates distributions of error free and error free runs that resemble those observed in experimental data. Another option for representing frame losses is to use experimental traces to drive simulations.

The two-state Markov model implementation is simple. Its characterization requires only two parameters. Its inclusion in a computer simulation is also uncomplicated since it only requires the computation of geometrically distributed variables to generate each state. During the good state all frames are assumed to arrive error free. During the bad state all frames are assumed to arrive in error.

A hidden Markov model implementation is more complex than a two-state Markov model since it not only includes the underlying Markov chain but a second Markovian process that during each state dictates how frames are lost. Its characterization requires matrix \mathbf{P} , vector $\boldsymbol{\pi}$ and the \mathbf{B} matrices. If a two-

state hidden Markov chain is used then a total of ten parameters are required. Its inclusion in a computer simulation requires the generation of both a Markov chain and an observable Markovian process.

The selection criteria between a two-state Markov model and a HMM should be based on how much accuracy is needed. For example, to study the transfer of files using TCP both models provided excellent results that were easy to validate because of the availability of experimental data. Therefore for such studies a two-state Markov model's accuracy is adequate. However, if a new protocol is studied, it is recommended to use hidden Markov models because these generate a frame process that adequately approximate the one observed experimentally. The dynamics of a new protocol could be affected by differences in the statistics of error or error free runs. Therefore a more accurate model like a HMM is required.

Two-state Markov model construction guidelines

1. The first step is to select the technology that the model should represent. Models for 802.11b and 802.11a technologies have different state parameter. Therefore one must first specify the technology.
2. No major differences in mean values or shapes of error and error free runs distributions between office and residential environments were noticed during the analysis. Therefore the choice of indoor environment to be considered is not critical. Most of the summary curves presented in this dissertation refer to an office environment. Hence for convenience, the results from this environment should be used to characterize indoor channels. This is valid for both 802.11b and 802.11a channels.
3. Since from the 2^k factorial design it was found that the frame size is the predominant factor that affects the duration of the error periods, the duration indicated by the smallest frame size should be taken as the duration for the corresponding state. The reason being that during the transmission of a large frame there is a higher probability of observing a series of changes in the channel state. During the transmission of a large frame the channel could go from "good" to "bad" and then back to the "good" state, however in the trace such a condition will be marked as "bad" during the whole transmission time. Using smaller frames gives better readings of the actual state of the channel. This is valid for both 802.11b and 802.11a channels.
4. The transmission rate and location selection define the values of the state durations needed for the model. Usually in 802.11b systems, simulations consider that transmission occur only at 11 Mbps. However this is not necessarily true due to the rate variation algorithms implemented in the systems. Therefore if a channel model for locations with high average SNR conditions is needed, high transmission rates should be considered. For low average SNR conditions, the lower transmission rates should be taken into account. With this in mind the next two tables

provide the information necessary to assign duration to the states in 802.11b and 802.11a systems under high, medium and low average SNR conditions. This information has been extracted from Figure 32 through Figure 38. The distinction between high, medium and low average SNR was based purely on the observed ordered range of average values of SNR.

Table 27 Two-state frame loss Markov model characterization parameters for 802.11b channels

SNR scale	Actual SNR considered	Transmission rate selected (Mbps)	Good state duration (seconds)	Bad state duration (seconds)
High SNR	32dB	11	0.08	0.0003
		5.5	4	0.0006
Medium SNR	23dB	5.5	2	0.00065
		2	3	0.001
Low SNR	12dB	2	0.02	0.0015
		1	0.035	0.0012

Table 28 Two-state frame loss Markov model characterization parameters for 802.11a channels

SNR scale	Actual SNR considered	Transmission rate selected (Mbps)	Good state duration (seconds)	Bad state duration (seconds)
High SNR	17dB	24	0.0015	0.00035
Medium SNR	14dB	12	0.8	0.0007
Low SNR	9dB	6	0.15	0.0012

To construct the model using the previous tables it is necessary to select both an SNR condition and a transmission rate and then read the values for the states' duration. Also notice that since the 2^k analysis of the experiments for 802.11a systems indicated that the site is a main factor in the duration of the states, Table 28 has only one rate per range of SNR.

Hidden Markov model construction guidelines

The step-by-step procedure to construct HMMs is very similar to the one that was proposed to construct the two-state Markov models. In the HMM case, different parameters should be characterized. To characterize a HMM like the one presented in Figure 17, the following procedure should be followed.

1. As with the two-state Markov model, the first step is to select the technology that the model should represent; this is either 802.11b or 802.11a.
2. As explained before, no major differences in mean values or shapes of error and error free runs distributions between office and residential environments were noticed during the analysis. Therefore the choice of indoor environment to be considered is not critical. Due to the availability of large amounts of data for office environments, results coming from these environments will be used to generate the models.
3. For the same reasons explained earlier on the guidelines for the construction of the two-state Markov model, one should select the results coming from small frame sizes traces to characterize the HMMs.
4. The transmission rate and location selection define the values of the state durations needed for the model. In a similar manner to that presented on the guidelines for the two-state Markov models, the next two tables provide the information necessary to construct a hidden Markov model such as that illustrated in Figure 17. The tables specify for each case the 2×2 transition probability matrix \mathbf{P} and two 2×2 matrices $\mathbf{B(0)}$ and $\mathbf{B(1)}$. The vector $\boldsymbol{\pi}$ can be computed by taking any row of the matrix \mathbf{P}^{10000} . After all these elements have been characterized it is possible to follow the approach from Section 3.10 to implement the model in a simulation.

Table 29 HMM frame loss characterization parameters for 802.11b channels

SNR scale	Actual SNR considered	Tx rate selected (Mbps)	P matrix		B(0) matrix		B(1) matrix	
High SNR	32dB	11	0.4570	0.5430	0.9751	0	0.0249	0
			0.1873	0.8127	0	0.9990	0	0.0010
		5.5	0.5975	0.4025	0.9997	0	0.0003	0
			0.0247	0.9753	0	0.9999	0	0.0001
Medium SNR	23dB	5.5	0.2937	0.7063	0.9998	0	0.0002	0
			0.1744	0.8256	0	1.0000	0	0
		2	0.9598	0.0402	0.9986	0	0.0014	0
			0.0116	0.9884	0	1.0000	0	0
Low SNR	12dB	2	0.6414	0.3586	0.6141	0	0.3859	0
			0.5168	0.4832	0	0.6721	0	0.3279
		1	0.8973	0.1027	0.3145	0	0.6855	0
			0.0434	0.9566	0	0.9655	0	0.0345

Table 30 HMM frame loss characterization parameters for 802.11a channels

SNR scale	Actual SNR considered	Tx rate selected (Mbps)	P matrix		B(0) matrix		B(1) matrix	
High SNR	32dB	17	0.8553	0.1447	0.9996	0	0.0004	0
			0.0749	0.9251	0	0.9995	0	0.0005
Medium SNR	23dB	14	0.9110	0.0890	0.9964	0	0.0036	0
			0.0700	0.9300	0	0.9998	0	0.0002
Low SNR	12dB	9	0.9182	0.0818	0.9903	0	0.0097	0
			0.1021	0.8979	0	1.0000	0	0

6.2. MODEL RESULTS AND VALIDATION OF THE TRANSMISSION RATE VARIATIONS

The transmission rate variations models were constructed using the methods detailed in Section 3.4. First HMM models were constructed using the RSSI as the variable for the “hidden” process, and then Markov models were tested and validated with K-S tests.

6.2.1. HMM and Markov model results for the representation of the transmission rate variations

HMM were constructed to represent the rate variations by taking the RSSI and partitioning it into two states and then following the same approach taken in Section 6.1. Before proceeding with the quantitative analysis and validation of the results it was noticed that the HMM was not generating adequate results. A simple visual validation was enough to determine the significant differences between the results.

Figure 49 shows a sample of the results for 802.11b systems. The figure illustrates a configuration in which 1,500 byte frames are being transmitted at location c. The top subfigure shows a sample of the evolution of the rate variations during the transfer. The two bottom subfigures show the results of modeling these variations with a HMM (left) and a four-state Markov model (right) characterized using the procedure described in Section 3.4.1. Figure 50 shows a sample of the results for an 802.11a transmission at the same location (location c at a SNR of 9dB at the receiver). For the 802.11a case the Markov model was constructed with eight states, one for each transmission rate available in the standard.

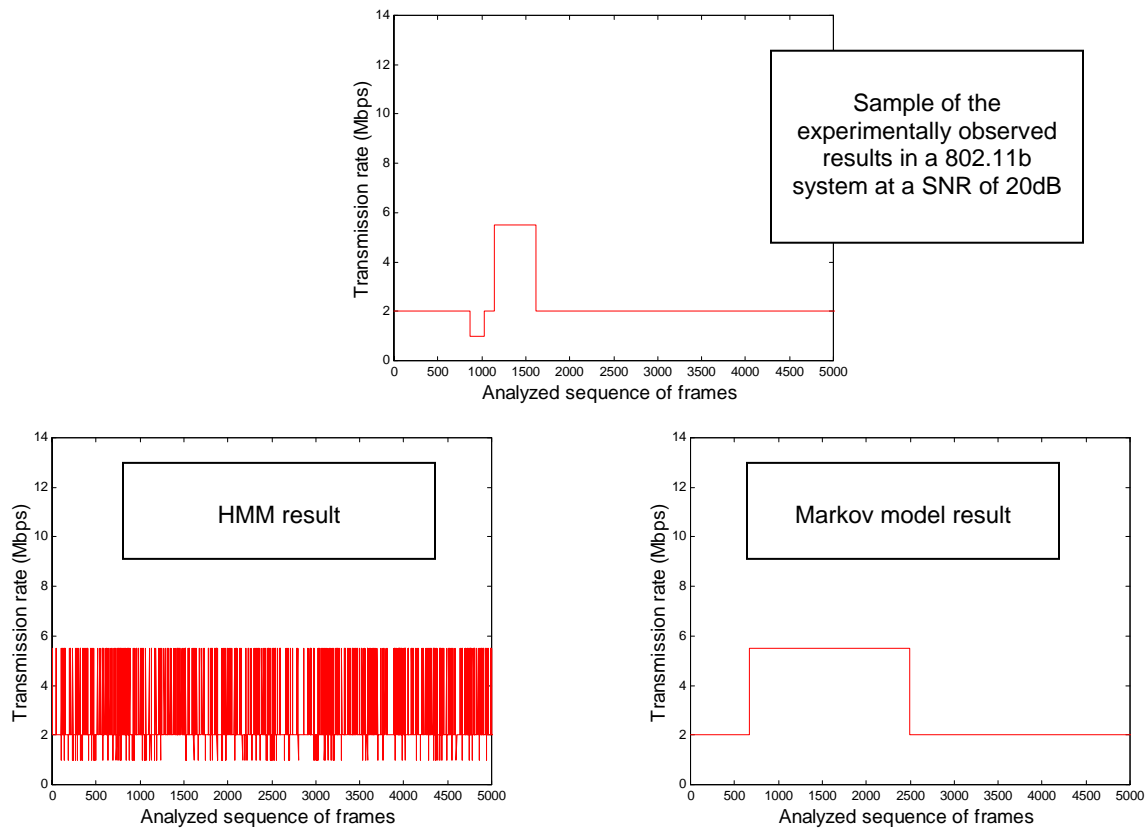


Figure 49 Sample transmission rate variations evolution during 5000 frames
(802.11b system at a SNR of 20dB at the receiver, location **c**)

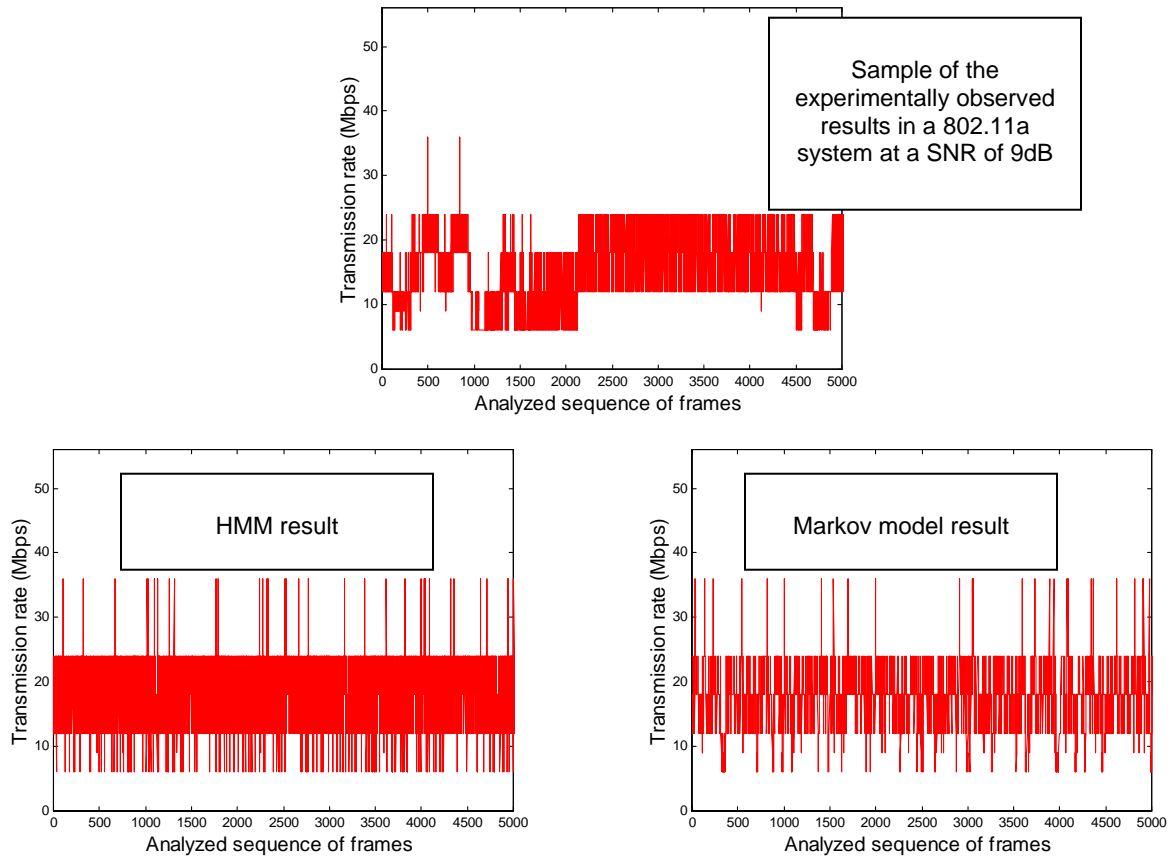


Figure 50 Sample transmission rate variations evolution during 5000 frames
(802.11a system at a SNR of 9dB at the receiver, location c)

A visual examination of these results indicates that the HMM is not able to follow the slow variations of the transmission rate and that the Markov model apparently outperforms it. However, further quantitative validation is necessary before the Markov models representation is accepted as valid.

6.2.2. Findings from the characterization of rate variations with a Markov model

A first step in the validation of the Markov model for the rate variations consisted in trying to fit known distributions to the experimental durations of each rate. Several distributions were tested using a one-sample K-S test but none of them offered an adequate fit. Since the distribution of the duration of each rate had an exponential like shape, a common four-state (for 802.11b) or eight-state (for 802.11a) Markov model was selected to characterize the variations. Then a two-sample K-S test was used to quantify the

differences between the distributions shapes. A sample of the results is presented in the following table. These results correspond to a setup that transfers 1500 byte frames at 27 dB (802.11b) and 9dB (802.11a). Other frame size and average SNR configurations were also tested yielding similar results.

Table 31 Two-sample K-S test results for the analysis of Markov models in transmission rate variations

Transmission (Mbps) rate from 802.11 systems	K-S Test shape result
1	>0.05
2	0.0145
5.5	0.0129
11	0.0107
Transmission (Mbps) rate from 802.11a systems	
6	>0.05
9	0.03
12	0.023
18	0.01
24	0.0013

The table indicates that the Markov model represent the rate duration distribution with low accuracy. This accuracy is observed to decrease as the transmission rate selected increases. However, the average transmission rate generated by a Markov model was statistically the same as that observed in the experimental traces. In particular for 1500 and 1000 byte transmissions the average modeled rate was in all cases less than one percent away from the ones observed in the traces.

6.2.3. Summary of findings and transmission rate variations modeling guidelines

The characterization of the transmission rate variations was approached with two different models. First a hidden Markov model was tested. Then and a four or eight-state Markov model was constructed for 802.11b and 802.11a technologies respectively. Experimental data was used to characterize both the Markov and the hidden Markov models.

Regarding the usage of the hidden Markov model:

- The results obtained from the hidden Markov model were highly inaccurate. There was no need to perform any quantitative analysis to reach this conclusion. A simple visual test strongly suggested that the model was not able to follow the very slow variations in the transmission rate.

- Even though the RSSI resulted in a valid variable to model frame losses, this was not the case with rate variations.
 1. In the case of frame losses any decrease or increase of RSSI results in an immediate change of the state of the channel. The “weak” Markovian properties of the RSSI were enough to avoid having noticeable negative effects on the accuracy of the hidden Markov models.
 2. In the case of rate variations a careful examination of the data from the traces revealed that variations in the RSSI were usually not followed by immediate changes of transmission rate. The rate changes are also a function of the actual proprietary algorithms implemented by the vendors in the wireless stations. Under these conditions the “weak” Markovian properties of the RSSI impeded the creation of an accurate hidden Markov model.
- The characterization of transmission rate variations with a finite state Markov chain approach results in a model that matches with high accuracy the average transmission rate and with a lower accuracy the distribution of the rate durations.
 1. In order to characterize a rate variations model, 1,500 byte frames were used. This is the frame size that wireless LAN stations will usually select to transmit information.

6.2.3.1. Model construction guidelines for rate variations

1. The first step is to select the technology that the model should represent. Models for 802.11b and 802.11a technologies have different state parameters and therefore, one needs to first specify the technology.
2. No major differences in mean values or shapes rate variations distributions between office and residential environments were noticed during the analysis. This is valid for both 802.11b and 802.11a channels. Therefore the choice of indoor environment to be considered is not critical. Due to the availability of large amounts of data for office environments, this environment will be used to generate the data in this section.
5. The 2^k factorial design indicated that the predominant factor that affects the rate variations is the location of the receiver and thus the average SNR at the receiver. With this in mind, the next two tables provide the information necessary to assign the transition probabilities from matrix P to a model similar to that shown in Figure 18 for 802.11b or 802.11a systems under high, medium and low average SNR conditions. The distinction between high, medium and low average SNR was based purely on the observed ordered range of average values of SNR. In the case of 802.11b the elements of this matrix represent the transitions between states that represent 1, 2, 5.5 and

11Mbps rates. In the case of 802.11a the elements of this matrix represent the transitions between states that represent 6, 9, 12, 18, 24, 36, 48 and 54 Mbps rates.

Table 32 Four-state rate variations Markov model characterization parameters for 802.11b channels

SNR scale	Actual SNR considered	Transition probability matrix P			
High SNR	32dB	0	1.0000	0	0
		0	0	1.0000	0
		0	0	0.9998	0.0002
		0	0	0.0003	0.9997
Medium SNR	23dB	0.9866	0.0134	0	0
		0.0001	0.9997	0.0002	0
		0	0.0030	0.9935	0.0035
		0	0	0.0023	0.9977
Low SNR	12dB	0.9950	0.0050	0	0
		0.0001	0.9997	0.0002	0
		0	0.0018	0.9982	0
		0	0	0	0

Table 33 Eight-state rate variations Markov model characterization parameters for 802.11a channels

SNR scale	Actual SNR	Transition probability matrix P							
High SNR	17dB	0.4376	0.5624	0	0	0	0	0	0
		0.4879	0.0184	0.4936	0	0	0	0	0
		0	0.0649	0.7206	0.2145	0	0	0	0
		0	0	0.1247	0.6891	0.1863	0	0	0
		0	0	0	0.1257	0.6989	0.1753	0	0
		0	0	0	0	0.1468	0.7976	0.0556	0
		0	0	0	0	0	0.2526	0.7402	0.0072
		0	0	0	0	0	0	0.2737	0.7263
Medium SNR	14dB	0.7453	0.2547	0	0	0	0	0	0
		0.4515	0.0720	0.4765	0	0	0	0	0
		0	0.1133	0.7336	0.1531	0	0	0	0
		0	0	0.1099	0.4837	0.4063	0	0	0
		0	0	0	0.0912	0.7407	0.1680	0	0
		0	0	0	0	0.0946	0.3944	0.5110	0
		0	0	0	0	0	0.2478	0.7173	0.0349
		0	0	0	0	0	0	0.7744	0.2256
Low SNR	9dB	0.6074	0.3926	0	0	0	0	0	0
		0.4770	0.0318	0.4911	0	0	0	0	0
		0	0.0630	0.6114	0.3256	0	0	0	0
		0	0	0.1225	0.7337	0.1437	0	0	0
		0	0	0	0.2489	0.7349	0.0162	0	0
		0	0	0	0	0.7727	0.2273	0	0
		0	0	0	0	0	0	0	0
		0	0	0	0	0	0	0	0

7. CONCLUSIONS

This dissertation has explored in diverse ways the characterization of 802.11 wireless channels. In a first step, the underlying fading phenomenon characterization was studied in order to further appreciate the challenges inherent to wireless channels in general. Fading in wireless channel has been widely studied and it is possible to use its characteristics to derive models that represent frame losses. A comprehensive review of how traditional models have been constructed in the past was given in order to understand the limitations present in those models. Using experimental data from actual 802.11 systems, models for frame losses and for the rate variations present in such systems were constructed and its results were validated at both the frame and transport levels. The main contribution developed throughout this dissertation has been the study of traditional models and the characterization of frame losses and rate variations in 802.11 systems, all of this performed with experimental data something that has not been done in the past.

Traditional frame loss models for systems like 802.11 have never been subjected to validation studies that indicate how well these models represent reality. This was a major task carried out in this dissertation and the findings were quite relevant. In the past, numerous research studies have used two-state Markov models to represent frame losses in wireless channels. With the experimental data collected for our studies it was determined that using the equations developed by Jakes [23] for the fading envelope results in a completely inaccurate representation of frame losses in 802.11 systems. The inadequate characterization of the model with Jakes' equations is due to the complexities inherent to 802.11 channels which use wideband channels with advanced modulation and coding schemes. Jakes' equations can not capture all these parameters and therefore are not useful for the generation of Markov models for 802.11 systems.

Furthermore, it is impossible to relate the parameters in Jakes' equations, such as the fading duration, average crossing rate or fade depth, with any of the commonly available variables in today's wireless networks. Variables such as the average signal to noise ratio, frame size or transmission speed are commonly used in current research studies because of the simplicity of the extraction of their actual values from a indoor wireless system. Having no way of relating these parameters with Jakes' variables

makes an accurate characterization of a Markov model for frame losses unfeasible. Any characterization with this kind of method was proven to be inadequate.

Frame losses in 802.11 wireless channels were adequately characterized with Markov models by using experimental data coming from actual measurements in indoor networks. It was found that using experimental data for constructing two-state Markov models for frame losses resulted in an accurate model. Two-state Markov models are simple to construct and provide a good match for the 802.11 frame loss process. However they do have a drawback. The distribution of consecutive periods of frames in error or error free do not resemble those observed in the experiments.

In order to further develop and test characterization methods that allowed the construction of accurate models, hidden Markov models were also tested. The results from HMMs were even better than those from the two-state model. HMMs provided with an accurate match of the underlying distribution of error and error free runs. A disadvantage of HMMs is that their construction requires more parameters than the two-state model, a fact that somewhat complicates its implementation.

Both two-state and HMM gave excellent results when their effects at the transport layer were measured. At this layer computer simulations that included TCP/IP communication between nodes, indicated that both models provide an adequate representation of reality. However if different protocols are to be tested it is recommended to use HMMs that more closely match the underlying frame loss process distributions.

Another major contribution of this dissertation is providing specific guidelines for the construction of both two-state and hidden Markov models. Chapter 6 presents all the necessary data to construct these models and use them in future simulation studies. Specific guidelines for both 802.11b and 802.11a systems allow the construction of models by relating the numerical results to common variables like the signal to noise ratio and the transmission speed. These types of guidelines, the author believes, are the first ones available in the subject. Furthermore, there are the only guidelines currently available that apply to 802.11 systems that use wideband channels which in general present challenges in their characterization.

A variable that has not been explored before, the transmission rate variation, was also studied. In the past research studies have not included the effects of the always present rate variations in 802.11 systems. These variations although they occur at slow rate (in comparison to the duration of a maximum sized frame) can definitely influence the performance of an 802.11 network. In this dissertation, models for the variations were constructed and specific guidelines for implementing them in future studies were provided.

The actual modeling of indoor 802.11 channels with Markovian approaches was developed in great detail in this dissertation. Nonetheless the rapid changes of today's technology provide new

research opportunities in the area. In particular it would be interesting to generate guidelines for other technologies such as 802.11g, multiple inputs/multiple outputs MIMO and sensor network systems. For all of these new technologies no guidelines are available today. In addition, the results from the experimental traces could also be used to further understand the behavior of the channel and perhaps predict its behavior.

APPENDIX A - ISSUES RELATED TO THE FSMC CHARACTERIZATION

The necessary elements needed to characterize a FSMC model were described in chapter two. Table 1 included a description of each of these elements. In summary these elements are the transition probability matrix \mathbf{P} , the steady state probability vector $\boldsymbol{\pi}$ and the crossover probability vector \mathbf{e} . This section will discuss several issues related to the characterization of these parameters. In particular issues regarding the partitioning methods of the SNR, the number of states, the modulation schemes and coding will be discussed.

A.1 CHARACTERIZATION OF \mathbf{P} , $\boldsymbol{\pi}$ AND \mathbf{e}

Under slow fading conditions, the channel is constant over the duration of a symbol and one can determine the transition probabilities between the states using the procedure suggested in [53]. When slow fading is present, the *level crossing rate* at any particular SNR partitioning level is very small compared to the total time spent in that state. Jakes [23] showed that given the maximum Doppler frequency (f_D) it is possible to compute the number of times (N_a) that the received SNR passes downward across a certain level ' a ' above or below the RMS value of the SNR.

The elements of \mathbf{P} can then be approximated by the relation between the crossing rate (N_a) and the 'symbol rate in each state,' R_k . R_k can be computed as the product (transmission rate $\times \pi_k$). For example the probability of going from state k to state $k-1$ is approximately equal to N_k/R_k . The accuracy of these approximations was verified via simulation [53].

Expressions for $\boldsymbol{\pi}$ and \mathbf{e} can be developed based on characteristics of the fading envelope and the modulation scheme. Wang and Moayeri [53] showed that with a pre-determined SNR partitioning scheme and assuming binary phase shift keying (BPSK) modulation is used, it is possible to obtain closed form expressions for the elements of the steady state probability vector $\boldsymbol{\pi}$ and the elements of the crossover probability vector \mathbf{e} . This can be done because the received signal envelope has a known probability distribution (i.e. Rayleigh distributed) and with this distribution one can compute the pdf of the received SNR. The pdf of the SNR in a Rayleigh fading environment with additive white Gaussian noise follows an exponential distribution [36].

Eventhough all three elements can be characterized using the guidelines given in [53], several open issues were still left for discussion. For instance, the way in which the SNR should be partitioned, the number of states, the order of the model and the relationship with the coding or modulation used are some issues that will be discussed in the next sections.

A.1. PARTITIONING

In order to construct the actual model, one of the parameters that must be specified is how the partitioning of the received SNR should be done. Wang and Moayeri [53] completely described the channel model but did not elaborate on the SNR partitioning. In the literature a common approach for the partitioning is to select the thresholds in such a way that the steady state probabilities of being in any state are equal [4][13][39][53]. In terms of the model parameters this means that the elements of the vector π can be expressed as follows,

$$\pi_0 = \pi_1 \dots \pi_{n-1} = 1/n. \quad (\text{for an 'n' state FSMC}) \quad (1)$$

This is referred as the *uniform partitioning scheme*. Right away one can appreciate the fact that the simplicity of this partitioning does not take into account the non-linearity between the SNR and the individual symbol probabilities of the BSCs of the model. Therefore, this partitioning scheme can be improved.

In addition to the simple partitioning method mentioned before two other partitioning schemes were proposed [52]. These schemes are derived from the quantization analysis for pulse code modulation techniques. The first scheme proposes making:

$$\pi_i = 2 \pi_{i-1} \text{ and } \pi_0 = 1/(2n-1). \quad (2)$$

This way the probability of being in a 'higher' level doubles as the state index 'i' increases. The second scheme proposes making:

$$\pi_i = i \pi_0 \text{ and } \pi_0 = 2/(n^2+n). \quad (3)$$

How these schemes are related to real channel conditions is not specified in [52]. There is also no comparison with simulated or real channel data.

The impact of these different partitioning schemes was evaluated in [52] by analyzing the *capacity* of the FSMC. The capacity measures how many bits per channel use can effectively be transmitted through the channel. The capacity computed is the *average capacity*, in bits per channel use, over all 'n' states. This is $\sum_{k=0}^{n-1} \pi_k (1 - h(e_k))$ where 'h' is the binary entropy function. Under average error probabilities ranging from 0.005 to 0.1 it was found that while keeping the number of states fixed, the

capacity differences between the two latter schemes were minimal (less than 1%). Greater differences in capacity were observed between the uniform partitioning scheme (1) and the ones given by (2) and (3). Nevertheless, the capacity is not directly related to the channel itself and thus to the accuracy of the model. Current literature however does not mention any other comparisons between the real channel behavior and the partitioning schemes.

A different approach for computing the partitioning thresholds was proposed by Zhang and Kassam [55]. Their goal was to select SNR partitioning intervals that are large enough so that a transmitted packet is completely received during each associated state. On the other hand in their approach, it is also desirable that within the duration of a packet similar SNR would be observed. That way, all bits in the packet will experience similar BER conditions. As shown in [55] the SNR interval cannot be made too large or too small but must be computed based on the packet duration. Zhang and Kassam formulated a system of linear equations that computes the duration of a state as a multiple of the duration of a packet. By solving this system of equations one can obtain the values for the SNR partitioning thresholds for a predetermined number of states. Once again, there is no validation with real or simulated channel data in [55]. There is also no discussion on how to select an appropriate number of states.

None of the previously mentioned partitioning schemes takes into account the fading characteristics of the channel. In [2] Aráuz and Krishnamurthy explored a new alternative. In that approach the authors studied the effect of the partitioning scheme on the model by placing a higher number of states in the regions where the average SNR value falls more often. Therefore, that approach necessarily depends on the average value of the SNR. The authors compared their scheme with others by looking at the same response variables used in early studies as well as some new response variables. The models were also validated using simulated channel data unlike some of the other approaches.

In order to take into account the channel fading characteristic in the model in [2] the authors proceeded to partition the received SNR based on the average fade duration at different levels of SNR. The relation between these two factors was determined by Jakes [23, pg. 37], with functions of the form

$\frac{e^{v^2} - 1}{v \times f_m \sqrt{k\pi}}$, where v is equal to the ratio R/R_{rms} and k depends on the field component that is computed.

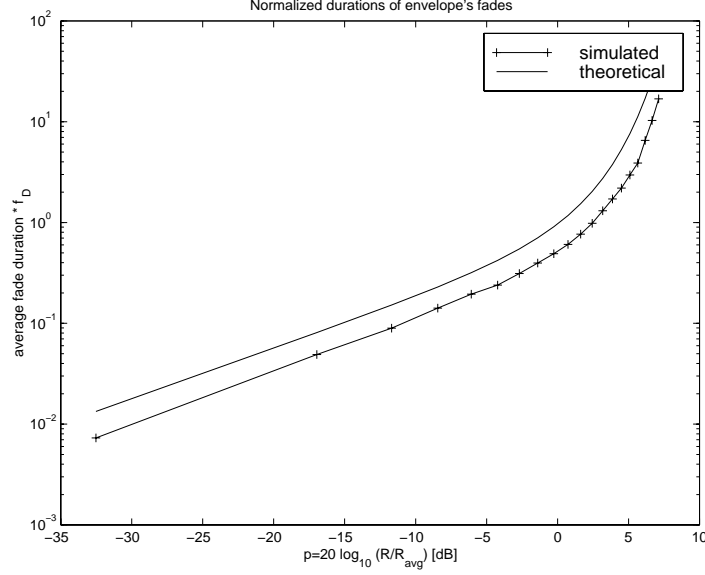


Figure 51 Normalized fade durations vs. R/R_{avg} [dB].

Figure 51 shows both the theoretical and simulated values obtained for the normalized duration of fades. The vertical axis plots the values (average fade duration $\times f_D$), while the horizontal axis plots the ratio between the level ‘R’ of the envelope and its average value ‘ R_{avg} ’. The figure shows how very deep fades last for very short periods of time. The theoretical results shown in the figure are computed based on the rms value of the envelope [23], while the simulated values are computed using the average value of the envelope. The simulated values and the results shown in this section were obtained using a modified version [35] of the sum of sinusoids fading simulator proposed by Jakes [23].

In [2] the SNR was partitioned in two new ways. Each of these two new schemes tries to study the effects of placing a higher number of states in regions of longer or shorter average fade duration. The first scheme includes 70% of the partitions over the mean value ($p=0$ [dB] in Figure 51) of the received SNR. The second one includes 70% of the partitions under this value. With knowledge of the SNR probability distribution one can compute the values of the elements of the vector π .

The width of the intervals located over the mean SNR value was computed as:

$$width = \frac{\max(r) - E[R]}{\lfloor m \times n \times 0.7 \rfloor}, \quad (4)$$

where ‘n’ is the number of states and ‘m’ was set to three to scale down the size of the intervals. If the value of ‘m’ is set to one (bigger intervals) the SNR does not fall often enough in the highest

partitions and this is not useful in practical implementations. The maximum value of SNR, $\max(r)$, is a known parameter from the simulation.

The values of the interval width below the mean are computed in a similar way (in this case ‘m’ is set to one):

$$width = \frac{E[R]}{\lfloor n \times 0.3 \rfloor}. \quad (5)$$

The second partitioning scheme proposed, uses equations similar to (4) and (5) but including 70% of the partitions under the mean value of the received SNR.

A partitioning scheme like the one proposed takes into account the average fade duration and this could be important in performance studies of higher layer protocols. These protocols transmit frames with several symbols over long periods of time that could span numerous fades.

For studying the effects of the partitioning scheme on the model one can look at the same response variables that have been previously analyzed [4][53][55]. In [2] the authors studied the state duration distribution, the elements of the transition probability matrix \mathbf{P} , the elements of the steady state vector $\boldsymbol{\pi}$ and the autocorrelation function.

The first two response variables are basically used to test the correctness and accuracy of the simulation. With regard to the elements of the matrix \mathbf{P} , the simulation results were compared with the mathematical approximations used in [53]. A χ^2 test was used to compare between the theoretical value of the elements of $\boldsymbol{\pi}$ and the observed ones in order to understand which partition results in a better fit.

In [2] five different partitioning schemes were studied; three previously proposed (see equations (1), (2), (3)) and those determined by equations (4) and (5). It was found that the observed values of the elements of the matrix \mathbf{P} were within 5% of the expected ones. The largest differences were found for the two highest states. This is because the highest states usually require more simulation samples to converge to the theoretical value. However the authors did not consider the accuracy of \mathbf{P} to be an adequate measure of fit for the partitioning scheme, since the obtained values mainly reflect the accuracy of the theoretical approximations for \mathbf{P} developed in [53].

In terms of the value of χ^2 for $\boldsymbol{\pi}$, it was found that the most commonly used scheme, the uniform partition approach, given by (1) gave the worst fit. The other four partitioning schemes (defined by equations 2 through 5) offered acceptable results. These results are shown in Table 34.

Table 34 χ^2 test for an 8, 16 and 32 state FSMC

Partition type	(1)	(2)	(3)	(4)	(5)
8 state	0.0033	0.0021	0.0029	0.0035	0.0022
16 state	0.0047	0.0021	0.0031	0.0038	0.0039
32 state	0.0057	0.0020	0.0034	0.0039	0.0039

From Table 34 it can be observed that the uniform partitioning scheme (1) has the worst fit in terms of matching the theoretical steady state probabilities. The second scheme offers an overall adequate fit to the simulated values. However for a high number of states this scheme is not practical since the interval widths for the highest states become very small and even very long simulations do not generate enough data to compute accurate statistics. More details are available in [2].

The other three schemes performed in similar ways. In the author's schemes, it was observed that taking the fading durations into account in (4) and (5) resulted in no considerable advantage in comparison to the other schemes although it is directly related to the fading process. The autocorrelation of the process was also used in validation of the models and those results are presented later in this chapter.

A.2. NUMBER OF STATES AND MODEL ORDER

The number of states that should be used in the FSMC model is another issue that has not been extensively studied. In general one could expect that a model with a higher number of states would represent the channel more accurately. Nevertheless, models with large number of states could be too complex for practical usage. There are several approaches that have been proposed to study the effects of different values for the number of states.

Babich and Lombardi [6] studied a two threshold (i.e. three states) FSMC model in a quantized Rayleigh fading environment. Based on experimental data they showed that a first-order FSMC (the first-order assumption will be discussed later) with three states gives a good approximation of the fading process under sufficiently slow fading ($f_m \times \tau < 0.02$). Under fast fading conditions ($f_m \times \tau > 0.4$), an uncorrelated model (zero-order model) proved to adequately approximate the fading process. For

intermediate values of fading Babich et al. [7][8] suggested a higher order model. Nevertheless, this literature does not elaborate extensively on the selection process of the number of states.

A.3. MODULATION AND CODING

As detailed elsewhere, the parameterization of the FSMC model requires the computation of the symbol error probabilities for the associated BSCs. Given a digital modulation scheme, the average error probability is a function of the received SNR, and it is possible to compute the crossover probabilities of each associated BSC. To do this, in general fairly simple modulation schemes have been used. For example [53] studied the behavior of the model using BPSK, while others have used $\pi/4$ differential quadrature phase shift keying (DQPSK) [55].

The modulation scheme could also have an effect on the statistics of block errors. Block errors are a function of the channel model and determine how packets are lost when transmitted over a wireless link. The modeling of the loss process is necessary when one studies the performance of upper layer protocols. Zorzi and Rao [56][57] studied the statistics of block errors when transmitting data over fading channels with different modulation schemes. In [56] it was found that a simple two state first order Markov model that describes the success/failure of transmitted blocks, called the *threshold model*, gives results that agree with those from a detailed symbol by symbol simulation (under slow fading conditions).

Using two modulation schemes, BPSK and frequency shift keying (FSK) Zorzi and Rao in [57] investigated the sensitivity of the block loss process to both the coding and modulation schemes. The block loss process was studied using the threshold model and the symbol by symbol model. At the block level, a block was considered in error when the value of the fading envelope is below a certain threshold (as also done in [56]). At the symbol level, a symbol was considered in error with a certain probability that depends on both the modulation scheme and the average SNR.

The effect of using a (N,k) block code was also included in [57]. A block was assumed to be correctly received when it contained fewer errors than those that the code is able correct. By tracking the fading envelope at the symbol level the authors included the effects of a varying envelope during the transmission of a block.

The results presented in [57] are quite interesting. It was found that a Markov approximation for the block loss process is a ‘very good model for a broad range of parameters’. For example, for block sizes (ranging from 100 to 2000 symbols per block), several error correcting capabilities and distinct levels of modulations, the authors showed [57, Fig. 5, 6] that the threshold model accurately approximates

the results obtained by the symbol by symbol tracking process. These results indicate that for tracking loss processes at the block level under very slow fading conditions, it is sufficient to use a two state first order Markov model (the threshold model). This model proves to be only sensitive to the value of f_D and not to the tested modulation or coding schemes.

APPENDIX B - VALIDITY AND ACCURACY OF THE FSMC

The most straightforward way of validating any model consists in comparing its results with experimental results. For example in the FSMC case, a comparison between the distributions of the time spent in each state could be performed between results from the model and results from experimental data taken from a sample function of the underlying random process. Previous literature mainly presents validations like this, but they usually use simulated and not experimental data to perform the comparisons [53][56]. Nevertheless, some results [6] also show that experimental channel data appears to be suitable for modeling with a Markov process.

In current literature, there are two widely studied validation approaches. The first type of validation is based on an information theory analysis. The second one compares the correlations of the processes under analysis.

Information Theory Validation Analysis - Before presenting the actual analysis for validating the FSMC, it is relevant to emphasize that this model by definition conforms to the Markov property. In particular, this property can be expressed as [26]:

$$p[S(t_n) = s_n | S(t_{n-1}) = s_{n-1}, S(t_{n-2}) = s_{n-2}, \dots, S(t_o) = s_o] = p[S(t_n) = s_n | S(t_{n-1}) = s_{n-1}]$$

This property indicates that the probability of transition at a time ‘n’ to a new state only depends on the state at time ‘n-1’ (also referred to as the *first-order assumption*). For the FSMC we are therefore assuming that the ‘history’ of previous channel states, besides the previous one, does not carry significant information about the next state. Without any further analysis it is difficult to visualize if the Rayleigh fading channel can be modeled following this assumption. Furthermore, it could appear to be more desirable to have a model that includes higher order assumptions and therefore maybe increase its accuracy [3][4]. However, the problem with higher order models is that the complexity of its analysis and implementation increases considerably.

Wang and Chang [54] proposed a ‘mutual information metric’ to verify the accuracy of the first order Markovian assumption for a Rayleigh FSMC model. The goal of the metric is to confirm that given the information about the previous symbol, the uncertainty of the current one should be negligible. This uncertainty is measured in terms of average mutual information of the received amplitudes.

Let A_i (where i is the time index) be the received SNR of the i^{th} symbol. The information contained in A_i given by the two consecutive (and previous) SNR values $A_{i-1}A_{i-2}$ is quantified by the average mutual information $I(A_i; A_{i-1}A_{i-2})$. As proposed in [21], this quantity can be expressed in terms of the average conditional mutual information $I(A_i; A_{i-2}|A_{i-1})$ and can be expressed as follows.

$$I(A_i; A_{i-1} A_{i-2}) = I(A_i; A_{i-1}) + I(A_i; A_{i-2} | A_{i-1}).$$

Wang and Chang's goal is to compute the value for the ratio $I(A_i; A_{i-2} | A_{i-1}) / I(A_i; A_{i-1} A_{i-2})$, which is a function of the joint pdf of A_i , A_{i-1} and A_{i-2} . Additionally, they showed how this pdf depends on the symbol transmission rate.

If the ratio $I(A_i; A_{i-2} | A_{i-1}) / I(A_i; A_{i-1} A_{i-2})$ is much smaller than one, then the average mutual information $I(A_i; A_{i-1} A_{i-2})$ mainly depends on the first term, $I(A_i; A_{i-1})$. This would mean that the information of A_i would mainly depend on the previous symbol A_{i-1} . If this happens, the first order assumption for the FSMC would be verified. Since the joint pdf of A_i , A_{i-1} and A_{i-2} depends on physical characteristics it is important to describe what these are and their ranges in order to maintain the FSMC validity.

The results presented in [54] show that for f_D ranging from 2×10^{-4} to 4×10^{-3} the value of the ratio $I(A_i; A_{i-2} | A_{i-1}) / I(A_i; A_{i-1} A_{i-2})$ is less than 1%. This value is even smaller for small values of f_D , since as fading gets slower the information of A_i is basically a function of A_{i-1} only. Therefore using higher order models will not improve the accuracy of the FSMC. On the other hand, for cases in which fast fading is observed the value of the ratio indicates that this is not negligible and the first order assumption is no longer valid.

Stochastic Validation Analysis - With the results presented in [54] the accuracy of the first order model is verified but nevertheless as indicated by Tan and Beaulieu [46] the fact that one has “small mutual information is not a sufficient condition to indicate a process is Markovian”. $I(A_i; A_{i-2} | A_{i-1})$ can actually approach zero in two cases. The first case is when the samples at i , $i-1$ and $i-2$ are independent and the second, when they are highly correlated. Under very slow fading conditions, such as those explored in [54], the samples are highly correlated.

Tan and Beaulieu [46] indicated that an appropriate way of verifying the accuracy of the first order Markovian assumption with information theory concepts is to analyze $I(A_i; A_{i-1}, A_{i-2}, A_{i-3}, \dots, A_\infty)$. The original validation in which Wang and Chang [54] analyze the value of $I(A_i; A_{i-2} | A_{i-1})$ only indicates that a second order Markovian model is marginally better than a first order one, but does not indicate that even higher order models are not better than the first order one. However, the intractability of the joint pdf needed for the analysis motivates the use of a different method. This second method was named “stochastic” analysis [46].

In the stochastic analysis the autocorrelation functions of the FSMC model and an ISORA model (isotropic scattering, omni directional receiving antenna [46]) are compared. The comparison of the autocorrelation functions of these two models provides an insight on how well the FSMC matches a generic ‘real’ model, the ISORA model. The results presented in [46, Fig. 2, 3] indicate that the autocorrelation function of a FSMC in general significantly differs from the ISORA model. These

differences between the autocorrelation functions are more noticeable as the fading rate increases. For example, at a value of f_D of 0.002 the two models appear to be more consistent with one another, whereas at a value of f_D of 0.02 the differences are quite noticeable. Additionally, from [46] it can be inferred that for slow fading conditions the two autocorrelation functions tend to match each other as the number of states of the FSMC increases. The range for the number of states used in [46] varied from 50 to 1000 states.

The autocorrelation functions shown by Tan and Beaulieu [46] suggested that the FSMC model is appropriate for very slowly fading channels but only for ‘very slowly fading’ applications. Very slowly fading applications are those that require analysis over a short duration of time. An example of a very slowly fading application could be the analysis of error correction code block-error rates, which according to [46] requires analysis over a moderate number of consecutive samples. By analyzing how the autocorrelation functions diverge over an increasing separation between sampling points, [46] arrives to the conclusion of how the FSMC is valid for very slowly fading applications. For very slowly fading applications, the autocorrelation functions of both the ISORA and the FSMC are very similar for distinct values of sample separations [46, Fig. 5].

Having described the accuracy of a FSMC model under very slowly fading conditions one can portray what happens under fast fading conditions. Under fast fading the autocorrelation function of the FSMC and the ISORA model differ but both tend to approach the conditions of an uncorrelated model over any fixed sample separation. This leads to the conclusion that an uncorrelated model is suitable under fast fading conditions. The details of how this model should be formulated are not given in [46] although it is pointed out that the implementation and analysis is much simpler than that of the FSMC.

The author also looked at the autocorrelation function of the FSMC model with the five different partitioning schemes detailed in the partitioning section. The approach used in [46] to compute the autocorrelation function of the schemes was implemented. The autocorrelation function R_R was computed as:

$$R_R[m] = \sum_{j=0}^{n-1} r_j \pi_j \sum_{i=0}^{n-1} r_i p_{i,j}^{(m)},$$

where m is the sample separation and the values of r_i were chosen to be the mid point of each of the partitioning intervals.

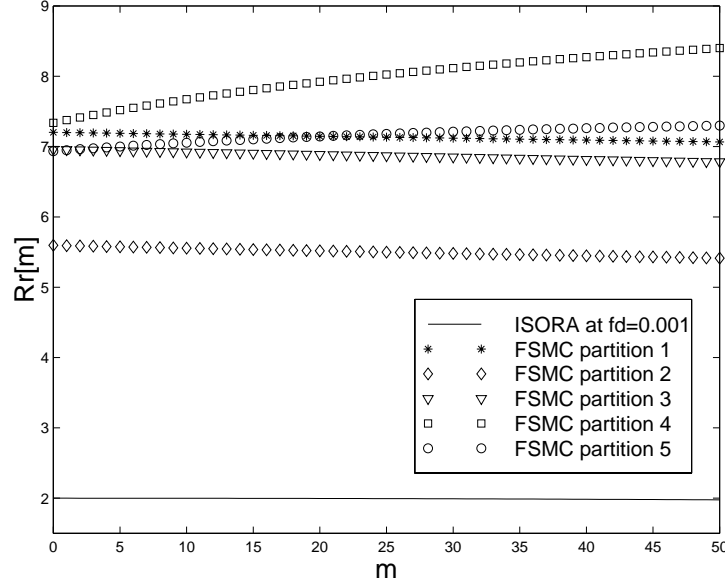


Figure 52 Autocorrelation functions R_R for the ISORA and FSMC models with 16 states at $f_D=0.001$.

Figure 52 shows the results obtained for the autocorrelation functions for the five partitioning schemes defined by equations one through five. It also shows the autocorrelation functions of the ISORA reference model for several values of sample separation ‘m’. The results shown are for slow fading conditions at value of f_D of 0.001.

From the results it can be observed that none of the partitioning schemes match the ISORA autocorrelation function. It is important to point out that it was found that the autocorrelation function values are very sensitive to the choice of elements to represent each interval (we selected the mid point of each interval). A slight variation in how the selection of these values is made can result in large variations of $R_R[m]$. It is even possible to closely approximate the theoretical ISORA autocorrelation function by selecting slightly different values. *Therefore, we do not consider this to be a good measure of fit for the FSMC model.*

Recently, Bergamo et al. [13] proposed an improvement for the approximation of the fading process with Markov models. Since a one-dimensional model like the FSMC does not show an ACF that is like the ISORA model, a two-dimensional model was suggested. This model takes into account not only the amplitude of the received SNR but its speed of variation. This new model is similar to the FSMC but also takes into account the difference in amplitude of two consecutive fading samples of the envelope. This difference is called ‘variation speed’.

In the two dimensional Markov model besides the set of states S it is necessary to define a set of ‘quantized’ variations speeds V . Therefore the transition matrix instead of being of order S^2 , as in the FSMC, is of order S^2V^2 . In the practical implementation of [13], V was chosen to have three states while the number of states was varied between two and 30. By taking into account the variation speed in the model the resulting ACF was shown to closely resemble the one from Jakes’ model [23]. Furthermore the two dimensional model was found insensitive to the number of states. This work however is still related only to narrowband flat fading channels.

APPENDIX C – FINITE STATE CHANNEL NOTATION

Let S represent the state space of a finite state channel and defined as $S=\{1, 2, \dots, u\}$. Each element of S represents an individual state of the channel. The transitions between states follow a sequence also known as regime. If the set of input symbols to the channel is $A=\{\mathbf{a}_1, \mathbf{a}_2, \dots\}$ and the set of output symbols is $H=\{\mathbf{h}_1, \mathbf{h}_2, \dots\}$ then the probability of being in state s_{t-1} with \mathbf{a}_t as input to the channel and going to state s_t with \mathbf{h}_t as the output can be written as $\Pr(\mathbf{h}_t, s_t | \mathbf{a}_t, s_{t-1})$.

Now let's suppose that the channel is in the initial state s_0 and after a series of transitions the final state of the channel is s_t . The sequence of observed outputs during this transitions is $\mathbf{h}^t=(\mathbf{h}_1, \mathbf{h}_2, \dots, \mathbf{h}_t)$ while the transmitted sequence of inputs was $\mathbf{a}^t=(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_t)$. We are interested in computing the probability of observing such sequence of outputs and ending in state s_t given the inputs and the initial state s_0 , that is we are interested in computing: $\Pr(\mathbf{h}^t, s_t | \mathbf{a}^t, s_0)$. This can be computed by taking into account every possible combination of sequence of states, outputs and inputs and can be expressed as [48 Eq. 1.1.1]:

$$\Pr(\mathbf{h}^t, s_t | \mathbf{a}^t, s_0) = \sum_{s_1^{t-1}} \prod_{i=1}^t \Pr(\mathbf{h}_i, s_i | \mathbf{a}_i, s_{i-1}) \quad (6)$$

This notation is too complex to be easily managed; therefore it is convenient to write the equations in matrix form. For this it is necessary to introduce the concept of matrix probabilities.

Let $\mathbf{P}(\mathbf{h}_t|\mathbf{a}_t)$ be a conditional matrix probability (MP) of dimension $u \times u$ of the output symbol \mathbf{h}_t given the input symbol \mathbf{a}_t . Each element (i,j) represents the probability $\Pr(\mathbf{h}_t, s_t=j | \mathbf{a}_t, s_{t-1}=i)$; this is the probability of transitioning from state i to state j while observing \mathbf{h}_t as the output of the channel with \mathbf{a}_t as the input. In general:

$$\mathbf{P}(\mathbf{h}_t|\mathbf{a}_t) = [\Pr(\mathbf{h}_t, s_t | \mathbf{a}_t, s_{t-1})]_{u,u}$$

With this definition (6) can be computed with the help of the expression:

$$\mathbf{P}(\mathbf{h}^t | \mathbf{a}^t) = \mathbf{P}(\mathbf{h}_1 | \mathbf{a}_1) \mathbf{P}(\mathbf{h}_2 | \mathbf{a}_2) \dots \mathbf{P}(\mathbf{h}_t | \mathbf{a}_t) = \prod_{i=1}^t \mathbf{P}(\mathbf{h}_i | \mathbf{a}_i),$$

where $\mathbf{P}(\mathbf{h}^t | \mathbf{a}^t)$ is a matrix whose (i,j) element is $\Pr(\mathbf{h}^t, s_t=j | \mathbf{a}^t, s_0=i)$. This matrix is called the *conditional MP of the output sequence given the input sequence* [48]. This notation will help in making expressions like (6) a lot easier to handle.

Let's see how these expressions can be used in an example. Suppose we have only two states, represented by state space $S = \{\alpha_1, \alpha_2\}$ (notice that for this example, α 's are used to represent the states). Additionally, let's suppose that the input and output sequences are of length 2 and represented by:

$$\mathbf{a}_1^{t=2} = (\mathbf{a}_1, \mathbf{a}_2)$$

$$\mathbf{h}_1^{t=2} = (\mathbf{h}_1, \mathbf{h}_2)$$

Then the conditional matrix probability $\mathbf{P}(\mathbf{h}_1|\mathbf{a}_1)$ can be expressed as 2×2 matrix that contains all the combinations of sequence of states that could have occurred while observing \mathbf{h}_1 at the output and \mathbf{a}_1 at the input. Let the state transition sequence for this example be represented by $(s_0) \rightarrow (s_I) \rightarrow (s_2)$, where (s_i) can be any of the α_i states, then the matrix corresponding to the state transition from $(s_0) \rightarrow (s_I)$ can be expanded as follows:

$$\mathbf{P}(\mathbf{h}_1 | \mathbf{a}_1) = \begin{pmatrix} \Pr(\mathbf{h}_1, s_1 = \alpha_1 | \mathbf{a}_1, s_0 = \alpha_1) & \Pr(\mathbf{h}_1, s_1 = \alpha_2 | \mathbf{a}_1, s_0 = \alpha_1) \\ \Pr(\mathbf{h}_1, s_1 = \alpha_1 | \mathbf{a}_1, s_0 = \alpha_2) & \Pr(\mathbf{h}_1, s_1 = \alpha_2 | \mathbf{a}_1, s_0 = \alpha_2) \end{pmatrix}$$

From the matrix shown above it can be observed that the first row represents all the possible state sequences that start in state α_1 and end either in state α_1 (first column) or α_2 (second column). The second row represents all the possible state sequences that start in state α_2 and end either in state α_1 (first column) or α_2 (second column).

In the same way for the next state transition, from $(s_I) \rightarrow (s_2)$ the matrix $\mathbf{P}(\mathbf{h}_2|\mathbf{a}_2)$ can be constructed as:

$$\mathbf{P}(\mathbf{h}_2 | \mathbf{a}_2) = \begin{pmatrix} \Pr(\mathbf{h}_2, s_2 = \alpha_1 | \mathbf{a}_2, s_1 = \alpha_1) & \Pr(\mathbf{h}_2, s_2 = \alpha_2 | \mathbf{a}_2, s_1 = \alpha_1) \\ \Pr(\mathbf{h}_2, s_2 = \alpha_1 | \mathbf{a}_2, s_1 = \alpha_2) & \Pr(\mathbf{h}_2, s_2 = \alpha_2 | \mathbf{a}_2, s_1 = \alpha_2) \end{pmatrix}$$

Then the matrix $\mathbf{P}(\mathbf{h}^2_1 | \mathbf{a}^2_1)$ can be obtained by multiplying $\mathbf{P}(\mathbf{h}_1 | \mathbf{a}_1) \times \mathbf{P}(\mathbf{h}_2 | \mathbf{a}_2)$. This matrix, $\mathbf{P}(\mathbf{h}^2_1 | \mathbf{a}^2_1)$ will be a 2×2 matrix in which all the possible combinations of state sequences are taken into consideration. For example in $\mathbf{P}(\mathbf{h}^2_1 | \mathbf{a}^2_1)$, the first element of the second row will represent the probability of observing the sequence of outputs and inputs and a sequence of states $(s_0 = \alpha_2) \rightarrow (s_I = \alpha_1) \rightarrow (s_2 = \alpha_1)$ plus the probability of observing the sequence of outputs and inputs with the state sequence

$(s_0=\alpha_2) \rightarrow (s_1=\alpha_2) \rightarrow (s_2=\alpha_1)$. Notice that the row index (2) indicates the initial state (α_2 in this case), while the column index (1) indicates the final state (α_1 in this case).

If one multiplies $\mathbf{P}(\mathbf{h}_1^t | \mathbf{a}_1^t)$ by a column vector of ones ($\mathbf{1}$) we obtain the vector $\mathbf{p}(\mathbf{h}_1^t | \mathbf{a}_1^t)$, which represents the probability of observing the output sequence conditioned on the input sequence for each initial state.

$$\mathbf{p}(\mathbf{h}_1^t | \mathbf{a}_1^t) = [\Pr(\mathbf{h}_1^t | \mathbf{a}_1^t, s_o=1) \quad \Pr(\mathbf{h}_1^t | \mathbf{a}_1^t, s_o=2) \quad \dots \quad \Pr(\mathbf{h}_1^t | \mathbf{a}_1^t, s_o=u)]$$

Therefore (6) $\Pr(\mathbf{h}_1^t, s_t | \mathbf{a}_1^t, s_o) = \sum_{s_1^{t-1}} \prod_{i=1}^t \Pr(\mathbf{h}_i, s_i | \mathbf{a}_i, s_{i-1})$ can now be expressed as:

$$\Pr(\mathbf{h}_1^t | \mathbf{a}_1^t) = \boldsymbol{\pi} \mathbf{p}(\mathbf{h}_1^t | \mathbf{a}_1^t) = \boldsymbol{\pi} \prod_{i=1}^t \mathbf{P}(\mathbf{h}_i | \mathbf{a}_i) \mathbf{1} \quad (7)$$

Where, $\boldsymbol{\pi} = [\Pr(s_o=1) \quad \Pr(s_o=2) \quad \dots \quad \Pr(s_o=u)]$ is the initial state probabilities vector.

APPENDIX D – PROBLEMS RELATED TO HMM

There are three main problems that are related to HMMs and their characterization. For stating the problems let's suppose a HMM is described by the set $\{S, H, \boldsymbol{\pi}, \mathbf{P}, \mathbf{B}(\mathbf{h})\}$, where S is the state space, H is the set of outputs, $\boldsymbol{\pi}$ is the initial probability vector, \mathbf{P} is the state transition probability matrix and $\mathbf{B}(\mathbf{h})$ are matrices whose diagonal elements are of the form $\text{diag}\{\Pr(\mathbf{h} \mid \text{state}=j)\}$, where \mathbf{h} belongs to the set of observations. The elements of a specific matrix \mathbf{B} for a particular observed output symbol 'h' at a given state s_j can be represented as $b_j(\mathbf{h})=\Pr(\mathbf{h}|s_j)$

Following the traditional notation let's write the vector $\boldsymbol{\pi}$ as, $\boldsymbol{\pi} = [\pi_1, \pi_2, \dots, \pi_N]$

D. 1. FIRST PROBLEM

Given an observation sequence O of output symbols, such that $O=O_1, O_2, \dots, O_T$, where $(O_i \in H)$ and a model $\boldsymbol{\theta} = (\boldsymbol{\pi}, \mathbf{P}, \mathbf{B}(\mathbf{h}))$. How do we efficiently compute $P(O \mid \boldsymbol{\theta})$?

A general solution will consist in enumerating every possible sequence of states of length T , one of these sequences could be $Q=q_1, q_2, \dots, q_T$, where $q_i \in S$ and q_1 is the initial state. Then the probability of the observation sequence O given the model $\boldsymbol{\theta}$ is obtained by summing the joint probability over all possible state sequences Q , this is:
$$P(O \mid \boldsymbol{\theta}) = \sum_{all_Q} P(O \mid Q, \boldsymbol{\theta}) P(Q \mid \boldsymbol{\theta}).$$

The number of operations for computing this probability is $(2T-1)N^T$ multiplications and (N^T-1) additions. Notice that even for 'small' models ($N < 10$) this results in an impractical number of computations. A reduction in the number of operations can be achieved by using the *forward-backward* algorithm.

Forward algorithm - The algorithm is based on the definition of a forward variable $\alpha_t(i)$, defined as the probability of observing a partial sequence O_1, O_2, \dots, O_t and being in state s_i at time t (given the model).

Therefore $\alpha_t(i) = P(O_1, O_2, \dots, O_t, q_t=s_i \mid \boldsymbol{\theta})$

The goal is to solve for $\alpha_t(i)$ by induction.

Induction process to solve for: $\alpha_t(i)$:

1. Initialization:

$$\alpha_1(i) = \pi_i b_j(O_1), \text{ for } 1 \leq i \leq N$$

where $b_j(O_1)$ is obtained from a matrix \mathbf{B} as the probability of observing O_1 and being in state i .

2. Induction:

In the induction phase we want to solve for the next $\alpha_t(i)$ (that is $t=t+1$) based on previous known values of α . Remember $\alpha_t(i)$ represents the partial observation probability up to time t . Now, let's suppose that at time $t+1$ state s_j is reached. This state can be reached from any of the previous s_1, s_2, \dots, s_N states. This is illustrated in the next figure:

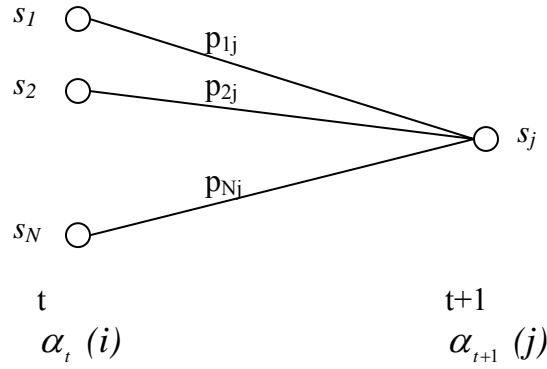


Figure 53 Forward algorithm next state representation diagram for the induction step.

Then,

$$\alpha_{t+1}(j) = \left(\sum_{i=1}^N \alpha_t(i) p_{ij} \right) \times b_j(O_{t+1})$$

3. Termination

The probability $P(O|\theta)$ is obtained by adding up all the α_T , then:

$$P(O|\theta) = \sum_{i=1}^N \alpha_T(i)$$

This algorithm requires, $N(N+1)(T-1)+N$ multiplications, $N(N-1)(T-1)$ additions. This results in a great improvement compared to the case in which all sequences are individually considered.

Backward algorithm - In a similar way that α was defined one can define a backward variable $\beta_t(i)$ that represents the probability of a partial observation sequence from time $t+1$ until time T , given that at time t the state was s_i , and given the model. Let's illustrate the general procedure.

The backward variable can be expressed as:

$$\beta_t(i) = P(O_{t+1}, O_{t+2}, \dots, O_T \mid q_t = s_i, \theta)$$

In the same way one can solve for $\beta_t(i)$ with induction.

1. Initialization:

Arbitrarily set $\beta_T(i) = 1$, for $1 \leq i \leq N$

2. Induction:

$$\beta_t(i) = \sum_{j=1}^N (p_{ij} \times b_j(O_{t+1}) \times \beta_{t+1}(j)), \text{ for } t=T-1, T-2, \dots, 1 \text{ and } 1 \leq i \leq N$$

The induction step can be illustrated with the following figure:

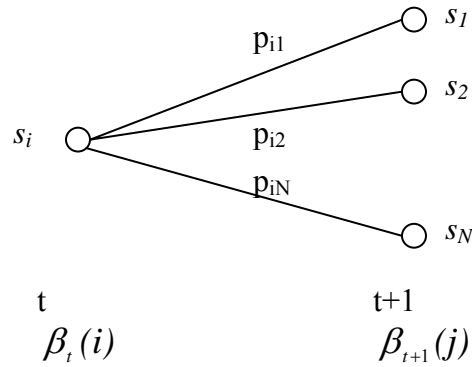


Figure 54 Backward algorithm previous state representation diagram for the induction step.

D.2. SECOND PROBLEM

Given an observation sequence O of output symbols, such that $O=O_1, O_2, \dots, O_T$ ($O_i \in H$), and the model $\theta=(\pi, P, B(h))$ we want to compute a corresponding state sequence which is “optimal” in some meaningful sense. Diverse optimality criteria have been defined, the most commonly used is the one that maximizes the probability $\Pr(Q|O, \theta)$, this can be computed via the Viterbi algorithm. For the work in this document, this method is not of interest since usually the states would represent signal to noise ratio variations. The research interests of this document do not lie on the evolution of the signal to noise ratio value but on the observable effects of these variations therefore we will not elaborate on the second problem.

D.3. THIRD PROBLEM

The third problem consists in determining the model parameters $(\pi, P, B(h))$ to maximize the probability of an observation sequence given a ‘first’ model.

There is no analytical solution to the problem of maximizing the probability, but there are methods to find locally maximized values of $P(O|\theta)$. For obtaining the local maximums one can use iterative procedures such as the Baum-Welch method (based on the EM method) or use gradient techniques.

The following discussion refers to the Baum Welch method:

Let’s define $\xi_t(i,j)$ as the probability of being in state s_i at time t and state s_j at time $t+1$, *given the model and the observation sequence*, then:

$$\xi_t(i,j)=P(q_t=s_i, q_{t+1}=s_j | O, \theta)$$

Using the forward and backward variables one can express $\xi_t(i,j)$ as:

$$\xi_t(i, j) = \frac{\alpha_t(i) \times p_{ij} \times b_j(O_{t+1}) \times \beta_{t+1}(j)}{P(O|\lambda)} = \frac{\alpha_t(i) \times p_{ij} \times b_j(O_{t+1}) \times \beta_{t+1}(j)}{\sum_{i=1}^N \sum_{j=1}^N \alpha_t(i) \times p_{ij} \times b_j(O_{t+1}) \times \beta_{t+1}(j)}$$

Where the denominator is included to make $\xi_t(i,j)$ a probability measure.

Now let’s define $\gamma_t(i)$ as the probability of being in state s_i at time t given the observation sequence and the model, this means:

$$\gamma_t(i) = P(q_t = s_i | O, \theta) = \frac{\alpha_t(i) \times \beta_t(i)}{P(O | \theta)} = \frac{\alpha_t(i) \times \beta_t(i)}{\sum_{j=1}^N \alpha_t(i) \times \beta_t(i)}$$

Where the denominator is included to make $\gamma_t(i)$ a probability measure, such that

$$\sum_{i=1}^N \gamma_t(i) = 1$$

One can relate $\gamma_t(i)$ and $\xi_t(i,j)$ by summing $\xi_t(i,j)$ over all possible j .

$$\gamma_t(i) = \sum_{j=1}^N \xi_t(i, j)$$

If one sums $\gamma_t(i)$ over the time index t , one obtains a quantity “that can be interpreted as the expected (over time) number of times that state s_i is visited, or equivalently the expected number of transitions made from state s_i (if we exclude the time slot $t=T$ from the summation, which would avoid taking into account the arbitrary value selected for $\beta_T(i)$ in the induction step). In a similar way summing $\xi_t(i,j)$ over t (from $t=1$ to $t=T-1$) can be interpreted as the expected number of transitions from state s_i to state s_j ”.

This means that:

$$\sum_{t=1}^{T-1} \gamma_t(i) = \text{expected number of transitions from } s_i.$$

$$\sum_{t=1}^{T-1} \xi_t(i, j) = \text{expected number of transitions from } s_i \text{ to } s_j.$$

With these expressions one can state a method for re-estimating the parameters of the HMM. It was proven by Baum that given the re-estimating formulas shown next, either the initial model defines a critical point of the likelihood function or the re-estimated model is more likely than the original one.

1. $\bar{\pi}_i = \text{expected frequency in state } s_i \text{ at time } t=1, = \gamma_1(i)$
2. $\bar{p}_{ij} = \frac{\text{expected number of transitions from state } s_i \text{ to state } s_j}{\text{expected number of transitions from state } s_i}$

$$\overline{p_{ij}} = \frac{\sum_{t=1}^{T-1} \xi_t(i, j)}{\sum_{t=1}^{T-1} \gamma_t(i)}$$

$$3. \overline{b_j(k)} = \frac{\text{expected number of times in state } s_j \text{ and observing symbol } a_k}{\text{expected number of times in state } s_j}$$

$$\overline{b_j(k)} = \frac{\sum_{t=1}^{T-1} \gamma_t(j)}{\sum_{t=1}^T \gamma_t(j)} \quad \text{s.t. } O_t = a_k$$

Based on these three expressions if one iteratively uses $\overline{\theta}$ instead of θ and repeat the procedure, one can improve the probability of O being observed from the model until a limiting point is reached in which the probability of observing O has been maximized locally. The final result of the procedure is called a maximum likelihood estimate of the HMM.

APPENDIX E – FURTHER SIMULATION DETAILS

The simulation results presented in this dissertation were generated with two tools. Matlab was used both for data analysis and simulation. Opnet Modeler was used for the implementation of Markov and HMM in environments that used UDP or TCP as transport media in 802.11 wireless networks. In Chapter 3 some relevant details on the Matlab implementation were given. This appendix discusses further details on the Modeler implementation.

E.1. THE WIRELESS CHANNEL MODEL IN OPNET'S MODELER

Modeler implements an advanced channel model for its wireless modules. The channel model is able to effectively handle collisions occurring in wireless networks by keeping track of the SNR conditions at each receiving station. The channel is represented by a series of stages that model an aspect of the overall behavior. Each state is implemented as a separate sequence of C or C++ procedures. For each of the stages a different pipeline model allows the representation of point-to-point, bus or radio channel conditions.

In wireless links the radio pipeline model is used. This pipeline consists of a series of stages that every transmission in the network follows. These stages compute for each frame the effects of transmission delay, frequency matching, propagation delay, antenna gain, background noise, bit error rate, SNR, interference, error allocation and error correction. At the receiver all of these effects will determine if a packet is received in error or error free. When several stations are configured in a wireless network in Modeler, the simulator can accurately keep track of when to discard packets at the receivers.

However, from the list of effects taken into consideration in the radio pipeline it should be clear that fading is not considered. This can be clearly verified by analyzing the percentage of frames lost when only two wireless stations are included in a network in Modeler. No frames are lost. The results presented throughout this dissertation have precisely shown that this is not what happens under such conditions in real networks.

To modify Modeler in a way that the fading effects are considered the simulator was altered. Since adding a new radio pipeline to those already available resulted quite a challenge, a different approach was taken. The alteration took advantage of a particular characteristic of the pipelines, this is that the pipeline stages are not in charge of discarding packets but only of informing the receiving station

of packets in error. Therefore the simulator was modified in a way that the fading effect was included at the moment packets are accepted or rejected at the MAC layer.

Packets are accepted or discarded by Modeler's wireless station during the execution of the wlan_mac process model. This process model is written in C and is in charge of implementing all the 802.11b standard MAC level details.

In particular, Modeler was modified as follows:

- New process models called fsmc or hmm were created to generate the transitions between a two-state Markov model or a HMM respectively.
- These process models were connected via a statistic wire to the wlan_mac process model. A statistic wire is a resource that allows inter-process communication. In this case the wire indicated the MAC layer if the channel was in a good or a bad state.
- The wlan_mac process model was modified at the point it discards packets. The information coming for the pipeline stages was not considered. Packets were discarded according to the information present in the statistic wire. The code modified in this process model corresponds to the wlan_physical_layer_data_arrival() function that is included (partially) next:

```
static void
wlan_physical_layer_data_arrival ()
{
    char                msg_string [120];
    int                 dest_addr;
    int                 accept;
    int                 final_dest_addr;
    int                 data_pkt_id;
    OpT_Packet_Id       pk_dhstruct_ptr;
    WlanT_Data_Header_Fields* pk_chstruct_ptr;
    WlanT_Control_Header_Fields* rcvd_frame_type;
    WlanT_Mac_Frame_Type wlan_rcvd_frame_ptr;
    Packet*             seg_pkptr;
    Packet*             pk_bbstruct_ptr;
    WlanT_Beacon_Body_Fields* rcvd_sta_bssid;
    int                 temp;
    int                 temp2;
    Boolean              data_pkt_received=OPC_FALSE;
    double              tx_drate,rcvd_pk_size, rx_start_time;
    char                actual_frame_name [256];
    /*jnas variables */
    int                 fsmc_mandate;
    char                temp_string[90];

    /* jnas var */

    /** Process the frame received from the lower layer.          **/
    /** This routine decapsulate the frame and set appropriate    **/
    /** flags if the station needs to generate a response to the  **/
    /** received frame.                                           **/
    /**
    FIN (wlan_physical_layer_data_arrival ());

    /* Access received packet from the physical layer stream. */
    wlan_rcvd_frame_ptr = op_pk_get (i_strm);

    op_pk_nfd_access (wlan_rcvd_frame_ptr, "Accept", &accept);
```



```

/* jnas changes to modify the retrieved value of accept */
/* according to the fsmc */
jnas_total_frames_received++;
fsmc_mandate = op_stat_local_read (CHANNEL_STATUS);
if (fsmc_mandate == 0 )
{
accept = OPC_TRUE;
op_stat_write (frame_seq_handle, 0.0);
/* notice that negative logic is used for frame seq */
}
if (fsmc_mandate == 1 )
{
accept = OPC_FALSE;
op_stat_write (frame_seq_handle, 1.0);
jnas_total_frames_received_in_error++;
}
op_stat_write(percent_frames_rx_in_err_handle, (jnas_total_frames_received_in_err
or/jnas_total_frames_received*100));
/* if fsmc_mandate is 99 the other possible value */
/* then accept is not changed at all */
/* Only first time */
if (jnas_first_temp == 0)
{
jnas_first_temp = 1;
jnas_pointto = accept;
if (accept) jnas_count_noerr++;
if (!accept) jnas_count_err++;

/* check if fsmc is active or not

if ( op_stat_local_read(5) == -1 )
{
op_sim_message ("FSMC activated for station with", temp_string);
}
else
{
op_sim_message ("Frames will be lost according to pipeline for
station with ", temp_string);
}

*/

}
else {
if ( accept == jnas_pointto)
{
if (accept) jnas_count_noerr++;
if (!accept) jnas_count_err++;
}
else
{
if (jnas_pointto) {
op_stat_write (consec_faccp_handle, jnas_count_noerr);
/* notice the other counter (the one with error) is set to one */
jnas_count_noerr = 0;
jnas_count_err = 1;
}
if (!jnas_pointto) {
op_stat_write (consec_fdrops_handle, jnas_count_err);
jnas_count_err = 0;
jnas_count_noerr = 1;
}
jnas_pointto = accept;
}
} /* end if second time */

/* Getting frame control field and duration information from*/

```

```

/* the received packet.
 */
op_pk_nfd_access (wlan_rcvd_frame_ptr, "Type", &rcvd_frame_type);

/* If the packet is received while the station is in transmission, */
/* or if the packet is collided with another packet received or if */
/* the accept field is set to false, then the packet will not be */
/* processed and if needed the station will retransmit the packet. */
if ((wlan_flags->rcvd_bad_packet == OPC_TRUE) || (accept == OPC_FALSE) ||
    (wlan_flags->collided_packet == OPC_TRUE))...

```

The code implements the frame loss process by first reading the status of the channel with the following instruction:

```
fsmc_mandate = op_stat_local_read (CHANNEL_STATUS);
```

With this instruction the channel status is read from the statistic wire labeled CHANNEL_STATUS and it is stored in the variable fsmc_mandate.

The channel status is read each time a frame arrives at the MAC layer. Then depending on the value of fsmc_mandate the accept flag of each packet is changed. When the channel indicates that packets should be discarded the accept flag is assigned a Boolean false, otherwise it is assigned a Boolean true. The rest of the code shown enables the generation of adequate statistics for following the loss process.

APPENDIX F INTERARRIVAL TIMES BETWEEN FRAMES

In order to compute the actual average duration of a particular state given the histogram of either the error free or error runs, the interarrival time between frames is necessary. To compute the average duration of a state in seconds it is necessary to multiply the average duration in frames by the interarrival times between frames. This number varies according to the frame size, transmission speed and processing power of the wireless stations. The next table provides the average value observed during the experiments. This average value was computed by considering all the frames transmitted in a trace that consisted of approximately 100,000 frames. For any transmission speed, frame size and wireless stations the interarrival time between frames varied less than 1% throughout an entire 100,000 frames trace.

Table 35 Average interarrival times (in seconds) for various frame sizes

Technology	Transmission Speed (Mbps)	1,500 byte frames	1,000 byte frames	500 byte frames	100 byte frames
802.11b	11	0.0016	0.0013	0.0009	0.0006
	5.5	0.0028	0.0020	0.0013	0.0006
	2	0.0066	0.0046	0.0026	0.0009
	1	0.0126	0.0086	0.0047	0.0013
802.11a	24	0.0007	0.0006	0.0004	-
	12	0.0012	0.0009	0.0005	-
	6	0.0022	0.0015	0.0012	-

APPENDIX G RESIDENTIAL ENVIRONMENT DATA

The traces collected for the experiments presented in this dissertation referred to two types of environment. Most of the data presented in previous chapters corresponds to office environments. In this Appendix a summary of the information obtained at the residential site is given. The next table shows the average duration of the error and error free runs at residential locations **p** and **q**.

Table 36 Average error and error free run duration (in seconds) at residential locations **p** and **q**.

Location	Technology	Transmission Speed (Mbps)	Average duration of error free run	Average duration of error
P	802.11b	11	16.48	0.0016
		5.5	4.6	0.0028
		2	12.18	0.0066
		1	109.8	0.001
	802.11a	24	0.033	0.0007
		12	6.12	0.0013
		6	0.0016	0.015
Q	802.11b	11	41.5	0.0017
		5.5	0.14	0.00812
		2	1.56	0.00818
		1	546.6	0.001
	802.11a	24	0.002	0.007
		12	1.39	0.0014
		6	0.09	0.44

BIBLIOGRAPHY

- [1] A. Abdi, Correspondence from the IEEE Transactions on Vehicular Technology, Vol. 48, No. 5, pp. 1739, September 1999.
- [2] J. Araújo, P. Krishnamurthy, "A Study of Different Partitioning Schemes in First Order Markovian Models for Rayleigh Fading Channels", The 5th International Symposium on Wireless Personal Multimedia Communications, Vol. 1, pp. 277-28, 2002.
- [3] F. Babich, G. Lombardi, "A Markov Model for the Mobile Propagation Channel", IEEE Transactions on Vehicular Technology, Vol. 49, No. 1, pp. 63-73, January 2000.
- [4] F. Babich, G. Lombardi, "On Verifying a First-Order Markovian Model for the Multi-Threshold Success/Failure Process for Rayleigh Channel", PIMRC 97, Vol. 1, pp. 12-16, 1997.
- [5] F. Babich, G. Lombardi, "Statistical Analysis and Characterization of the Indoor Propagation Channel", IEEE Transactions on Communications, Vol. 48, No. 3, pp. 455-464, March 2000.
- [6] F. Babich, G. Lombardi, "A Measurement Based Markov Model for the Indoor Propagation Channel", IEEE 47th Vehicular Technology Conference, Vol. 1, pp. 77-81, 1997.
- [7] F. Babich, O. Kelly, G. Lombardi, "Generalized Markov Modeling for Flat Fading", IEEE Transactions on Communications", Vol. 48, No. 4, pp. 547-551, April 2000.
- [8] F. Babich, O. Kelly, G. Lombardi, "A Context-Tree Based Model for Quantized Fading", IEEE Communications Letters, Vol. 3, No. 2, pp. 46-48, February 1999.
- [9] A. Bateman, "A General Analysis of Bit Error Probability for Reference Based BPSK Mobile Data Transmissions", IEEE Transactions on Communications, Vol. 37, pp. 398-402, April 1989.
- [10] S. Bengio, Y. Bengio, "An EM Algorithm for Asynchronous Input/Output Hidden Markov Models", Proceedings of the International Conference on Neural Information Processing, Hong Kong, 1996.
- [11] Y. Bengio, V. Lauzon, R. Ducharme, "Experiments on the Application of IOHMMs to Model Financial Return Series", IEEE Transactions on Neural Networks, Vol. 12, No. 1, pp. 113-123, January 2001.
- [12] Y. Bengio, "Markovian Models for Sequential Data", Neural Computing Surveys, Vol. 2, pp. 129-162, 1999.
- [13] P. Bergamo, D. Maniezzo, A. Giovanardi, G. Mazzini, M. Zorzi, "An Improved Markov Chain Description for Fading Processes", ICC, pp. 1347-135, Vol. 3, 2002.
- [14] J. Bilmes, "A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models", International Computer Science Institute, April 1998.
- [15] R. Chandramouli, "A Stochastic Technique for On-line Prediction and Tracking of Wireless Packet Networks", Conference Record of the Thirty-Fifth Asilomar Conference on Signals, Systems and Computers, Vol. 1, pp., 672-676, 2001.
- [16] H. M. Chaskar, T.V. Lakshman, U. Madhow, "TCP Over Wireless with Link Level Error Control Analysis and Design Methodology", IEEE/ACM Transactions on Networking, Vol. 7, No. 5, pp. 605-615, October 1999.

- [17] M. Chiani, E. Milani, Verdone R., "A Semi-Analytical Approach for Performance Evaluation of TCP-IP Based Mobile Radio Links", IEEE Global Telecommunications Conference, pp. 937-942, Vol. 2, 2000.
- [18] S. Ci, H. Sharif, A. Young, "Frame Size Adaptation for Indoor Wireless Networks", Electronics Letters, Vol. 37, No. 18, pp. 1135-1136, August 2001.
- [19] G. Cobb, "Introduction to Design and Analysis of Experiments", New York, Springer-Verlag, 1998.
- [20] D. Collet, "Modeling Binary Data", Second Edition, Boca Raton, Chapman and Hall/CRC, 2003.
- [21] R. G. Gallager, "Information Theory and Reliable Communication", New York, Wiley, 1968.
- [22] J. Gómez, A. Campbell, "A Channel Predictor for Wireless Packet Network", IEEE International Conference on Multimedia and Expo, pp. 1269-1272, Vol. 3, 2000.
- [23] W.C. Jakes, "Microwave Mobile Communications", New York, McGraw Hill, 1989.
- [24] R. Jain, "The Art of Computer Systems Performance Analysis", New York, John Wiley and Sons, 1991.
- [25] J. G. Kim, M. Krunz, "Quality of Service Over Wireless ATM links", Proceedings of the IEEE Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies, pp. 1003-1010, Vol. 3, 1999.
- [26] L. Kleinrock, "Queueing Systems", Vol. 1, New York, John Wiley & Sons, 1975.
- [27] H. Kong, E. Shwedyk, "Markov Characterization of Frequency Selective Rayleigh Fading Channels", Proceedings of the IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing, pp. 359-362, 1995.
- [28] H. Labiod, "Performance of Reed Solomon Error-Correcting Codes on Fading Channels", IEEE International Conference on Personal Wireless Communication, pp. 259-263, 1999.
- [29] W. C. Y. Lee, "Mobile Communications Engineering", New York, McGraw Hill, 1992.
- [30] B. Lindgren, "Statistical Theory", Third Edition, New York, Macmillan Publishing Co., 1960.
- [31] R. W. Lucky, "Principles of Data Communication", New York, McGraw-Hill, 1968.
- [32] M. Mushkin, I. Bar-David, "Capacity and Coding for the Gilbert-Elliot Channels", IEEE Transactions on Information Theory, Vol. 35, No. 6, pp. 1277-1289, November 1989.
- [33] N. Nefedov, "Generative Markov Models for Discrete Channel Modeling", PIMRC 97, Vol. 1, pp. 7-11, 1997.
- [34] K. Pahlavan, P. Krishnamurthy, "Principles of Wireless Networks: A Unified Approach", Prentice Hall, 2001.
- [35] M. F. Pop, N. C. Beaulieu, "Limitations of Sum-of-Sinusoids Fading Channel Simulators", IEEE Transactions on Communications, Vol. 49, No. 4, pp. 699-708, April 2001.
- [36] J. G. Proakis, "Digital Communications", 2nd. Ed, New York, McGraw-Hill, 1989.
- [37] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition", Proceedings of the IEEE, Vol. 77, No. 2, pp. 257-286, February 1989.
- [38] T. Rappaport, "Wireless Communications", Upper Saddle River N.J, Prentice Hall, 1996.

- [39] C. Schuler, "Error Correction Strategies for Wireless ATM", The Fourth IEEE Workshop on High-Performance Communication Systems, pp. 204-213, 1997.
- [40] S. Sivaprakasam, K. S. Shanmugan, "An Equivalent Markov Model for Burst Errors in Digital Channels", IEEE Transactions on Communications, Vol. 43, No. 2/3/4, February/March/April 1995, pp. 1347-1355.
- [41] J. R. Shewchuck, "An Introduction to the Conjugate Gradient Method Without Agonizing Pain," Tech. Rep. CMU-CS-94-125, Carnegie Mellon University, 1994.
- [42] B. Sklar, "Rayleigh Fading Channels in Mobile Digital Communication Systems part I: Characterization", IEEE Communications Magazine, Vol. 35, Issue 7, pp. 90-100, July 1997.
- [43] B. Sklar, "Rayleigh Fading Channels in Mobile Digital Communication Systems part II: Mitigation", IEEE Communications Magazine, Vol. 35, Issue 7, pp. 102-109, July 1997.
- [44] F. Swarts, H. C. Ferreira, "Markov Characterization of Channels with Soft Decision Outputs", IEEE Transactions on Communications, Vol. 41, No. 5, pp. 678-682, May 1993.
- [45] J. Swarts, H. Ferreira, "On The Evaluation and Application of Markov Channel Models in Wireless Communications", IEEE Vehicular Technology Conference, Vol. 1, pp. 117-121, Fall 1999.
- [46] C. C. Tan, N. C. Beaulieu, "On First-Order Markov Modeling for the Rayleigh Fading Channel", IEEE Transactions on Communications, Vol. 48, No. 12, pp. 2032-2040, December 2000.
- [47] W. Turin, "Performance Analysis of Wireless Systems using Hidden Markov Models", IEEE Vehicular Technology Conference Fall 2001 Tutorial.
- [48] W. Turin, "Digital Transmission Systems", New York, McGraw-Hill, 1999.
- [49] W. Turin, M. M. Sondhi, "Modeling Error Sources in Digital Channels", IEEE Journal on Selected Areas in Communications, Vol. 11, No. 3, pp. 340-347, April 1993.
- [50] W. Turin, R. Van Nobelen, "Hidden Markov Modeling of Flat-Fading Channels", IEEE Journal on Selected Areas in Communications, Vol. 16, No. 9, pp. 1234-1238, December 1998.
- [51] W. Turin, M. Zorzi, "Performance Analysis of Delay-Constrained Communications Over Slow Rayleigh Fading Channels", IEEE Transactions on Wireless Communications, Vol. 1, No. 4, pp. 801-807, October 2002.
- [52] H. S. Wang, N. Moayeri, "Modeling, Capacity and Joint Source/Channel Coding of Rayleigh Fading Channels", 43rd IEEE Vehicular Technology Conference, pp. 473-479, 1993.
- [53] H. S. Wang, N. Moayeri, "Finite-State Markov Channel – A useful Model for Radio Communications Channels", IEEE Transactions on Vehicular Technology, Vol. 44, No. 1, pp. 163-171, February 1995.
- [54] H. S. Wang, P. Chang, "On Verifying the First-Order Markovian Assumption for a Rayleigh Fading Channel Model", IEEE Transactions on Vehicular Technology, Vol. 45, No. 2, pp. 353-357, May 1996.
- [55] Q. Zhang, S. A. Kassam, "Finite-State Markov Model for Rayleigh Fading Channels", IEEE Transactions on Communications, Vol. 47, No. 11, pp. 1688-1692, November 1999.
- [56] M. Zorzi, R. R. Rao, L. B. Milstein, "On the Accuracy of a First-order Markov Model for Data Transmission on Fading Channels", 4th IEEE International Conference on Universal Personal Communications, pp. 211-215, 1995.

- [57] M. Zorzi, R. R. Rao, L. B. Milstein, "Error Statistics in Data Transmission over Fading Channels", IEEE Transactions on Communications, Vol. 46, No. 11, pp. 1468-1476, November 1998.
- [58] M. Zorzi, R. R. Rao, "ARQ Error Control for Delay-Constrained Communications on Short-Range Burst-Error Channels", IEEE 47th Vehicular Technology Conference, Vol. 3, 1997, pp. 1528-1532.
- [59] M. Zorzi, R. R. Rao, "Throughput Analysis of Go-Back-N ARQ in Markov channels With Unreliable Feedback", IEEE ICC, Vol. 2, pp. 1232-1237, 1995.
- [60] M. Zorzi, R. R. Rao, "On the Statistics of Block Errors in Bursty Channels", IEEE Transactions on Communications, Vol. 45, No. 6, pp. 660-666, June 1997.